Deep learning for remote sensing images and their interpretation

Ines Meraoumia

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2023IPPAT050

Thèse de doctorat

Deep learning for remote sensing images and their interpretation

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (ED IP PARIS)
Spécialité de doctorat : Signal, Images, Automatique et Robotique

Thèse présentée et soutenue à Palaiseau, le 14/12/2023, par

INÈS MERAOUMIA

Composition du Jury :

Julie Delon Professeure, Université Paris Cité, IUF	Présidente/Examinatrice
Emmanuel Trouvé Professeur, Université Savoie Mont Blanc, Polytech Annecy-Chambéry	Rapporteur
Thomas Oberlin Maitre de conférence, ISAE-SUPAERO, Université de Toulouse	Rapporteur
Andrea Marinoni Assistant Professor, UiT the Arctic University of Norway	Examinateur
Ronan Fablet Professeur, IMT Atlantique	Examinateur
Florence Tupin Professeure, Télécom Paris, Institut Polytechnique de Paris	Directrice de thèse
Loïc Denis Professeur, Université Jean Monnet Saint-Etienne	Co-directeur de thèse

J'ai soudain le sentiment étrange
d'être en harmonie avec moi-même,
tout est parfait en cet instant,
la douceur de la lumière,
ce petit parfum dans l'air,
la rumeur tranquille de la ville.
J'inspire profondément car la vie
me parait alors si simple,
qu'un élan d'amour
me donne tout à coup envie
d'aider l'humanité tout entière.

Le fabuleux destin d'Amélie Poulain.

Remerciements

Ce manuscrit résume mes travaux scientifiques des trois dernières années, mais il ne saurait rendre compte de l'intégralité des étapes que j'ai pu traverser pendant ma thèse. Il est difficile pour moi de résumer ces années, mais je peux tout de même décrire en quelques mots l'importance qu'elles ont eu pour moi. Elles ont été une mise à l'épreuve de ma confiance en moi, de ma rigueur et de ma détermination. Elles ont aussi été une aventure humaine et des rencontres de part le monde.

Pour ces souvenirs et ces compétences acquises, je tiens tout d'abord à remercier mes encadrants de thèse Florence et Loïc, vous avez été de véritables mentors pour moi.

Florence, merci pour ta gentillesse, ta disponibilité et l'accompagnement à la fois scientifique et humain que tu m'as offert. Tu as toujours su me motiver dans les moments difficiles.

Loïc, merci pour ta rigueur scientifique et de m'avoir poussée dans mes retranchements en me challengeant toujours plus. Malgré la distance, j'ai toujours eu l'impression que tu n'étais pas très loin. Je te remercie particulièrement pour ton accueil chaleureux pendant mes déplacements à Saint-Etienne.

Je remercie aussi les membres de mon jury: Julie Delon, Emmanuel Trouvé, Thomas Oberlain, Ronan Fablet, Andrea Marinoni et Rémy Abergel. Merci pour votre implication et le précieux temps que vous avez consacré à l'étude de mes travaux.

L'enseignement a fait partie intégrante de ma thèse et je tiens à remercier les enseignants chercheurs qui m'ont pris sous leurs ailes: merci à Saïd, Christophe, Philippe et Alasdair pour les cours d'OASIS, Florence à nouveau pour les cours d'IMA.

Le Deep Learning Working Group a aussi rythmé mes semaines au laboratoire, et pour ceci je tiens à remercier particulièrement Gwilherm d'avoir été mon partenaire d'organisation, mais je remercie aussi tous les doctorants qui ont finalement cédé à mes supplications incessantes pour présenter leurs travaux, ainsi que ceux qui ne l'ont pas fait.

Je tiens aussi à remercier Emanuele pour toutes nos collaborations, ainsi que les membres du projet PHC AURORA COSMIC pour leur accueil à Tromso durant 3 semaines de nuit polaire.

Je remercie aussi Zoé pour avoir fait de notre bureau un endroit joyeux et vivant, théâtre de longs débats en tous genres.

Pour leur soutien moral infaillible, je remercie Jade ma chère colloque, Victoire, Jeanne et Eugénie.

Merci à mes parents et à ma famille de m'avoir soutenue dans tous mes choix et d'avoir tout fait pour que je sois heureuse et fière de moi.

Et *last but not least*, je remercie Max, tu sais que cette aventure n'aurait pas été possible sans toi, sans ton soutien inconditionnel malgré 6 heures de décalage horaire.

Résumé

Le radar à synthèse d'ouverture (SAR) n'est pas impacté par la présence de nuages ou la luminosité et permet donc l'acquisition d'images riches en informations pour l'observation de la Terre (chapitre 1). De fortes fluctuations appelées "speckle" sont néanmoins visibles sur ces images et rendent leur interprétation difficile. Le "speckle" est un phénomène intrinsèque à l'illumination cohérente de la scène par le capteur est vient de la somme des échos des différents éléments au sol au sein d'une case radar. Des interférences constructives et destructives ont lieu et donnent naissance aux fluctuations appelées speckle. Des images sans fluctuation ne peuvent donc pas être acquises. Les propriétés du speckle sont différentes de celles du bruit additif blanc gaussien usuellement utilisé en imagerie optique. Les algorithmes de despeckling sont donc propres aux statistiques du speckle du modèle de Goodman (chapitre 2). Récemment, des méthodes d'apprentissage profond ont donné de très bons résultats pour la restauration d'une seule image SAR. Les travaux proposés au sein de cette thèse utilisent le traitement conjoint de plusieurs images SAR pour améliorer leur restauration en exploitant l'information commune tout en empêchant la propagation de potentielles différences (chapitre 3).

Le chapitre 4 est centré sur le despeckling des images Sentinel-1 GRDM Extra Wide de la glace de mer. La glace se déplaçant rapidement sur la mer, des changements structurels apparaissent rapidement sur une zone d'intérêt, rendant les piles multi-temporelles inexploitables. Le bruit thermique de ces images ne peut pas être négligé car les valeurs de réflectivité de l'eau et de la glace sont très faibles et proches du seuil du bruit thermique. Notre méthode de despeckling polarimétrique utilise et restaure conjointement les canaux polarimétriques HH et HV disponibles dans les données d'intérêt. Une zone du Nord de la Russie a été retenue pour l'entraînement du réseau et s'inspire de la méthode SAR2SAR, en prenant en entrée des images corrigées où la composante de bruit thermique a été supprimée. La qualité des images Sentinel-1 de l'Arctique restaurées avec notre approche est bien meilleure que celle obtenue avec d'autres techniques de restauration. Une piste pour la validation du débruitage via l'analyse des résultats d'une méthode de classification des différents types de glaces de mer est proposée en perspective.

Utiliser l'information partagée au sein d'une pile multi-temporelle tout en ignorant l'impact des changements temporels améliore le despeckling comme le montrent les travaux de débruitage multi-temporel des images SAR. Des méthodes de despeckling multitemporel basées sur un moyennage temporel ou l'utilisation d'une super-image construite à partir d'une moyenne temporelle débruitée sont d'abord présentées dans le chapitre 5. Un modèle génératif est ensuite proposé afin d'expliciter la formation d'une pile multi-temporelle d'images SAR en tenant compte des corrélations spatiales et temporelles du speckle. Une extension multitemporelle de la méthode MERLIN est basée sur ce modèle génératif et prend en entrée des images additionnelles de la même zone mais acquises à des dates différentes. L'entrainement du réseau est non supervisé et s'inspire de la méthode Noise2Noise : la partie réelle (ou la partie imaginaire)

de l'image et les dates additionnelles sont transmises au réseau et la partie imaginaire (ou la partie réelle) est utilisée comme cible. Un premier entraînement sur du speckle simulé montre que l'ajout d'images supplémentaires améliore la restauration des images SAR avec un gain décroissant. Un blanchiment temporel est proposé pour éviter une perte de performance liée aux corrélations temporelles entre les canaux d'entrée. L'entraînement du réseau a été effectué sur des images TerraSAR-X en modalité stripmap ainsi que des images Sentinel-1 en modalité stripmap.

L'absence d'image de référence rend l'évaluation des méthodes de despeckling difficile. Le chapitre 6 se concentre sur la quantification des incertitudes liées à la prédiction d'un réseau. Des travaux combinant le despeckling et l'estimation d'une carte d'incertitudes sont d'abord présentés. Dans le cadre d'origine, i.e. la méthode MERLIN, une seule valeur de réflectivité est prédite pour chaque pixel. Dans ces travaux d'estimation d'incertitudes, nous visons à prédire les paramètres d'une distribution choisie pour chaque pixel. Les paramètres des lois uniforme puis inverse gamma sont estimés lors de l'entraînement, mais les résultats trouvés ne sont pas concluants. La difficulté à quantifier les incertitudes dans un cadre d'apprentissage auto-supervisé où le niveau de bruit est élevé est discutée via une analyse de l'erreur relative dans un cadre plus simple de corruption par un bruit additif gaussien. Une autre méthode est proposée : en utilisant le principe du réseau MERLIN, la prédiction de la carte moyenne des différences entre les prédictions basées sur la partie réelle et imaginaire est prédite pas un autre réseau. Elle fournit une carte d'incertitudes satisfaisante.

Contents

Scalar and vector notations:		
j	$\mathbb{C}$	imaginary unit
$\boldsymbol{z}$	$\mathbb{C}^{TN}$	representation of a stack of T N-pixels images
$\boldsymbol{z}(\cdot, k)$	$\mathbb{C}^{T}$	vector of values at pixel k
$\boldsymbol{z}(t, \cdot)$	$\mathbb{C}^{N}$	t-th image of the stack
$\boldsymbol{z}_t$	$\mathbb{C}^{N}$	t-th image of the stack (compact notation)
$\boldsymbol{z}_{\text{ref}}$	$\mathbb{C}^{N}$	image at date t_{ref}, the date to restore
Scene parameters:		
$\boldsymbol{d}$	$\mathbb{C}^{TN}$	dominant scatterers
$\boldsymbol{r}$	$\mathbb{R}_{+*}^{TN}$	reflectivities of speckled areas
$\boldsymbol{r}_t$	$\mathbb{R}_{+*}^{N}$	t-th reflectivity image of the stack
$\boldsymbol{i}_{\text{super}}$	$\mathbb{R}_{+*}^{N}$	super-image computed with the T images of the stack
$\bar{\boldsymbol{i}}$	$\mathbb{R}_{+*}^{N}$	temporal mean (intensity) computed with the T images of the stack
Dual-polarimetric GRD images:		
$\boldsymbol{\Sigma}$	$\mathbb{C}^{2N\times 2N}$	polarimetric covariance matrix
$\widetilde{\boldsymbol{d}}$	$\mathbb{R}_{+*}^{2N}$	calibrated GRD intensities
$\widetilde{\boldsymbol{d}}_c$	$\mathbb{R}_{+*}^{2N}$	calibrated and corrected GRD intensities
$\boldsymbol{\sigma}^0$	$\mathbb{R}_{+*}^{2N}$	calibrated reflectivities
$\boldsymbol{\sigma}_{th}^0$	$\mathbb{R}_{+*}^{2N}$	thermal noise floor component
Speckle field:		
$\boldsymbol{\epsilon}$	$\mathbb{C}^{TN}$	uncorrelated speckle
$\boldsymbol{\Gamma}_k$	$\mathbb{C}^{T\times T}$	speckle coherence matrix at pixel k
$\mathbf{L}_k$	$\mathbb{C}^{T\times T}$	correlating operator such that $\mathbf{L}_k\mathbf{L}_k^\dagger = \boldsymbol{\Gamma}_k$
$\mathbf{L}$	$\mathbb{C}^{TN\times TN}$	correlating operator for the full stack
Complex amplitudes on the radar antenna:		
$\boldsymbol{s}$	$\mathbb{C}^{TN}$	complex amplitude of the speckled component
$\boldsymbol{z}$	$\mathbb{C}^{TN}$	resultant complex amplitude: $\boldsymbol{z} = \boldsymbol{s} + \boldsymbol{d}$
$\tilde{\boldsymbol{z}}$	$\mathbb{C}^{TN}$	complex amplitude including SAR system effects
Acquisition specific parameters:		
$\boldsymbol{\varphi}_t$	$\mathbb{C}^{N}$	atmospheric, topographic, and displacement phase effects at each pixel of the t-th image
$\boldsymbol{\psi}_t$	$\mathbb{C}^{N}$	phase ramp corresponding to the spectrum shift due to angular discrepancies
$\mathbf{Q}$	$\mathbb{C}^{N\times N}$	SAR response (spectral apodization and 0-padding)
$\mathbf{H}_t$	$\mathbb{C}^{N\times N}$	SAR response (spectral apodization, 0-padding+shift)
Pre-processing step to enforce statistic independence:		
$\dot{\boldsymbol{z}}$	$\mathbb{C}^{TN}$	complex amplitudes with recentered power spectrum
$\gamma_{ij}(k)$	$\mathbb{C}$	complex correlation coefficient (i.e., coherence) between $\dot{\boldsymbol{z}}(t_i, k)$ and $\dot{\boldsymbol{z}}(t_j, k)$
$\mathbf{W}_k$	$\mathbb{C}^{2\times 2}$	whitening matrix at pixel k
$\mathbf{W}$	$\mathbb{C}^{2N\times 2N}$	whitening operator for a pair of images
$\mathring{\boldsymbol{z}}$	$\mathbb{C}^{TN}$	complex amplitudes after whitening
Self-supervised training:		
$\mathring{\boldsymbol{a}}_{\text{ref}}$	$\mathbb{C}^{N}$	real part of pre-processed image at date t_{ref}
$\mathring{\boldsymbol{b}}_{\text{ref}}$	$\mathbb{C}^{N}$	imaginary part of pre-processed image at date t_{ref}
$\mathcal{L}_{\text{MERLIN}}$		self-supervised loss function
$\tilde{\boldsymbol{r}}_{\text{ref}}$	$\mathbb{R}_{+*}^{N}$	low-pass filtered reflectivities at date t_{ref}
$\dot{\boldsymbol{d}}_{\text{ref}}$	$\mathbb{C}^{N}$	low-pass filtered dominant scatterers at date t_{ref}

Table 1: Main notations and corresponding dimensions.

Part I

Content of the work and research directions

Chapter 1

Background and main objectives of the thesis

1.1 Introduction to Remote-Sensing and its challenges from an image processing perspective

1.1.1 Overview of Remote-sensing: different ways to acquire images of the Earth

NASA defines remote-sensing as *the acquiring of information from a distance*. Remote-sensing is used to observe the Earth with sensors onboard satellites or aircrafts, aiming at detecting the reflected or emitted energy of various objects on the ground.
Airborne sensors are usually used to monitor a very specific area and produce very high resolution images. Because of their relatively small distance to the ground, there is limited impact of the atmosphere on the resulting image. However, the geometry of the acquisition is more complex because the aircraft is less stable and its trajectory can be winding.
Remote-sensing satellites acquire images of the Earth from space. Their orbits can be classified in three main categories: low-Earth orbit (from 160 to 2000 km above the ground), medium-Earth orbit (from 2000 to 35 500 km above the ground); and high-Earth orbit (more than 35 500 km above the ground). The orbits can be geostationary (so the satellite keeps seeing the same area on Earth) or follow various orbital tracks (most of the time from pole to pole, leading to cycles of a fixed duration to scan the entire planet).

Different kinds of sensors are used in remote-sensing, each one belonging to two main categories as shown in Figure 1.1. Passive sensors are based on the analysis of radiation emitted by the scene but coming from external sources; and active sensors are based on the analysis of radiation scattered back by the scene coming originally from the sensor itself.
Passive sensors are capturing the power of the radiated light by the objects on the ground. The acquisition mechanism is the same as a camera, and the sensor is targeting the ground with an incident angle almost corresponding to nadir. The sensor can be sensitive to different wavelengths: optical images are obtained with visible wavelengths (400nm $\leq \lambda \leq$ 800nm); multi-spectral and hyperspectral images are also acquired with infrared radiations (from 800nm $\leq \lambda \leq$ 12 500nm). The

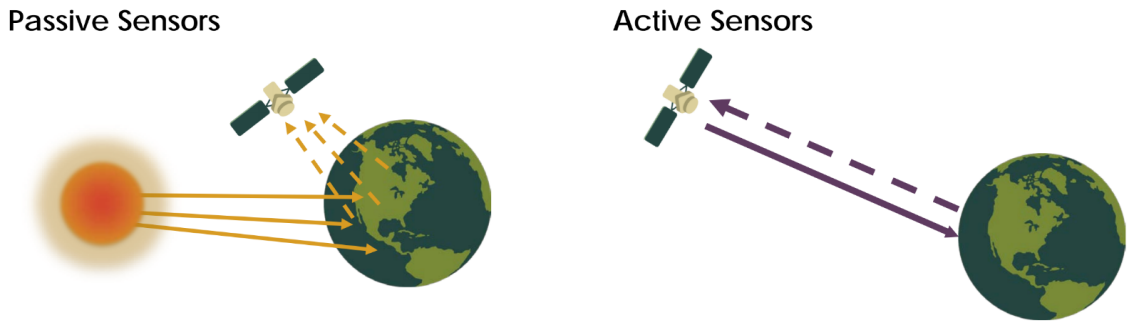


Figure 1.1: Passive sensors: radiation emitted by the sun is reflected by the objects on Earth. Their radiation is captured by the sensors. Active sensors: a radiation is emitted by the sensor and scattered back by the objects on the scene. ©NASA Applied Sciences Remote Sensing Training Program.

wavelength used to monitor a region is chosen based on what we want to study. Red radiations are used to observe human structures whereas mid-infrared and even far-infrared radiations are used to observe vegetation and forests, and are commonly used for military application.
These kinds of images, and especially optical images, are very easy to interpret and correspond to the general idea of satellite imaging. Their characteristics are close to the ones of natural images, making the use of traditional methods easy. Unfortunately, optical images can not be interpreted when the cloud coverage is high: as the sensor is taking a *picture* of the scene, clouds and even a lack of light (night for example) will lead to unreliable images. Monitoring equatorial climate zones is thus very challenging or even impossible depending on the period of the year because of a very cloudy and rainy weather. An example of the evolution of cloud coverage is given in Figure 1.2 for the city of La Paz in Bolivia. Another extreme case could be for areas near the North or the South poles where the nights can last up to 3 months. In the archipelago of Svalbard in Norway, the Polar night starts on November 11th and ends on January 30th. During this time, light intensity is low and monitoring sea ice or icebergs in the Arctic is almost impossible with optical images.

Active sensors like radars are emitting a radiation toward the area of interest. On the ground, a first part of the radiation is absorbed, then a second part is reflected and the final part is scattered. The backscattered radiation is captured by the sensor and an image can thus be reconstructed. The image is directly linked to the structures and the materials on the ground. Because of the wavelengths used in radars, the acquisition does not rely on the weather and the atmospheric conditions have a very low impact (this will shortly be discussed in this manuscript in Chapter 5). Thus, the monitoring of an area on the ground or oceans can be done at all time, during night and day, and through all seasons.

Nowadays, numerous satellites are gravitating around the Earth, each one having its specificity and dedicated fields of applications. A non-exhaustive list is given in Table 1.1. In this work, we will mainly focus on *active* sensors and more specifically Synthetic Aperture Radar such as **Sentinel-1** and **TerraSAR-X** satellites.

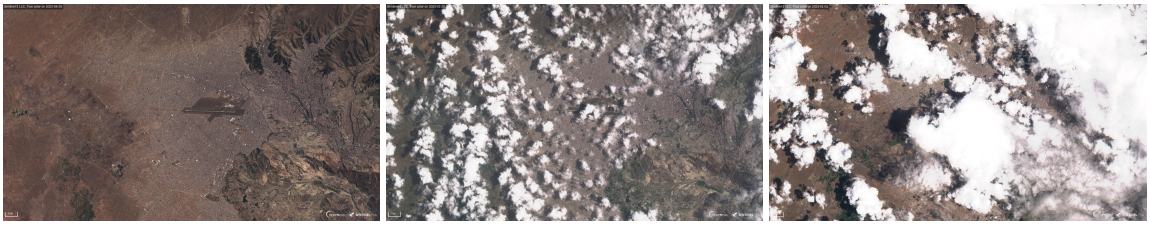


Figure 1.2: Sentinel 2 optical images of the city of La Paz, Bolivia. First image on the left was acquired on 29/08/2023, cloud coverage equal to 0%; the second (central) image was acquired on 20/02/2023 with a cloud coverage equal to 17%; last image on the right was acquired on 01/01/2023 with a cloud coverage equal to 47%. Even if the central image has a medium cloud coverage, we can see that it is very difficult to perform segmentation or detection on this urban area. The last image is impossible to process. In practice, the cloud parts of the images are ignored when processed. The retrieval of the data needs a threshold percentage for the cloud coverage which is fixed depending on the application.

1.1.2 RADAR and SAR images

RADAR (RAdio Detection and Ranging) is the most notorious active system used in remote-sensing, and has lead to the Synthetic Aperture Radar imaging. The general concept of this type of imaging relies on the timing of the propagation of an electromagnetic radiation emitted by the sensor. Part of the radiation is scattered back to the sensor itself and based on the received echoes, an image is produced.
The acquisition of Synthetic Aperture Radar images is carried with a satellite or an airborne which is side-looking at the ground. They are generated using several acquisitions during the Synthesis step: because the resolution of the image depends on the size of the antenna, a synthetic antenna is simulated by acquiring multiple images of the same area and combining them. A focusing along the range (direction orthogonal of the displacement of the sensor, associated with the abscissa in the Cartesian plane of the image) and then on the azimuth (direction of displacement of the sensor, associated with the ordinate in the Cartesian plane of the image) is done as shown in Figure 1.3. Because the bandwidth of the sensor is limited, a convolution by a sinc function for each pixel of the image is visible and the signature of each scatterer is wide-spread. To reduce the sidelobes and make the SAR image more interpretable, a 0-padding and an apodization function is applied to the spectrum of the SAR image. The synthesis step is not detailed in this manuscript but further details are given in [1].

The final images are complex-valued and give an information in amplitude and also in phase. Because we are working with electromagnetic radiations, different polarizations can be observed depending on the sensor. Most of the time, we have the HH (horizontal transmission, horizontal reception) and VV (vertical transmission, vertical reception) polarizations, and the cross polarizations HV (horizontal transmission, vertical reception) and VH (vertical transmission, horizontal reception). Polarimetric information will be used in Chapter 4 of this manuscript.

Because the sensor is side looking and has an incident angle from 15 to 50 degrees, we can observe geometrical deformations as explained in Figure 1.4. These deformations are not easy to interpret as shown in Figure 1.5.

Moreover, SAR images are corrupted by a noise called *speckle*. The speckle is a multiplicative

Name	Type of sensor	Spatial agencies involved	Resolution	Year of launching	Revisit cycle
RADARSAT-2	SAR	CSA (Canadian Space Agency) MacDonald Dettwiler Associates Ltd. of Richmond, BC	1-100m	2007	24 days
TerraSAR-X	SAR	DLR (German Aerospace Center)	0.5-16m	2007	11 days
COSMO-SkyMed (4 satellites)	SAR	Italian Space Agency Thales Alenia Space	1-100m	2007/2007 2008/2010	16 days
Pléiades 1A/1B	Optical and infrared	CNES (Centre National d'études spatiales, France) Airbus Defense & Space Thales Alenia Space	0.7m	2011/2012	1 day
ALOS-2 Advanced Land Observing Satellite-2	SAR	JAXA (Japan Aerospace Exploration Agency)	3-100m	2014	14 days
Sentinel-1 A/B	SAR	ESA (European Space Agency)	5-40m	2014/2016	6 days
Sentinel-2 A/B	Multi-spectral sensor	ESA	10-60m	2015/2017	5 days
RADARSAT Constellation (3 satellites)	SAR	CSA (Canadian Space Agency)	1-50m	2019	1 day
Landsat-9	Optical and thermal	NASA USGS (United States Geological Survey)	15-30m	2021	16 days

Table 1.1: Non-exhaustive list of active satellites in orbit nowadays. The resolution varies in a wide range because of the different modes used by the sensor to acquire images. In this thesis, we have mainly worked with Sentinel-1 and TerraSAR-X images whose characteristics have been highlighted in the table.

noise. Its level is very high, leading to many difficulties to process SAR images. Speckle is directly linked to the physics of the scene: it results from the mixing of different echoes from different objects on the floor within one resolution cell. Adding all the echoes leads to constructive and destructive interferences and thus a corruption of the original scene. Besides, because of the apodization of the spectrum, the information in one pixel is spread across its neighbors and thus the speckle is spatially correlated. Further details on the speckle characteristics and statistics will be given in section 2.2.

When dealing with SAR images, many products exist. Because the resolution of SAR images depends on the number of sensor locations combined to form the synthetic antenna, the *mode* of the sensor plays an important role. If it is doing multiple acquisitions of an area by overlapping scans, there will be more images to combine and the resulting SAR image will have a better resolution. Other pre-processing can be done to the product to ease interpretation. Two kinds of products will be used in this thesis:

- Single Look Complex Images: they have complex values. We can use properties related to the amplitude, the phase, and their polarimetric information.
- Ground Range Detected images: they are processed by the provider to have square pixels on the ground. To verify this property, a multi-looking (spatial averaging of pixel intensities) needs to be done. This multi-looking is not isotropic on range and azimuth. The speckle on these images is reduced thanks to the spatial averaging, and only the intensity values are

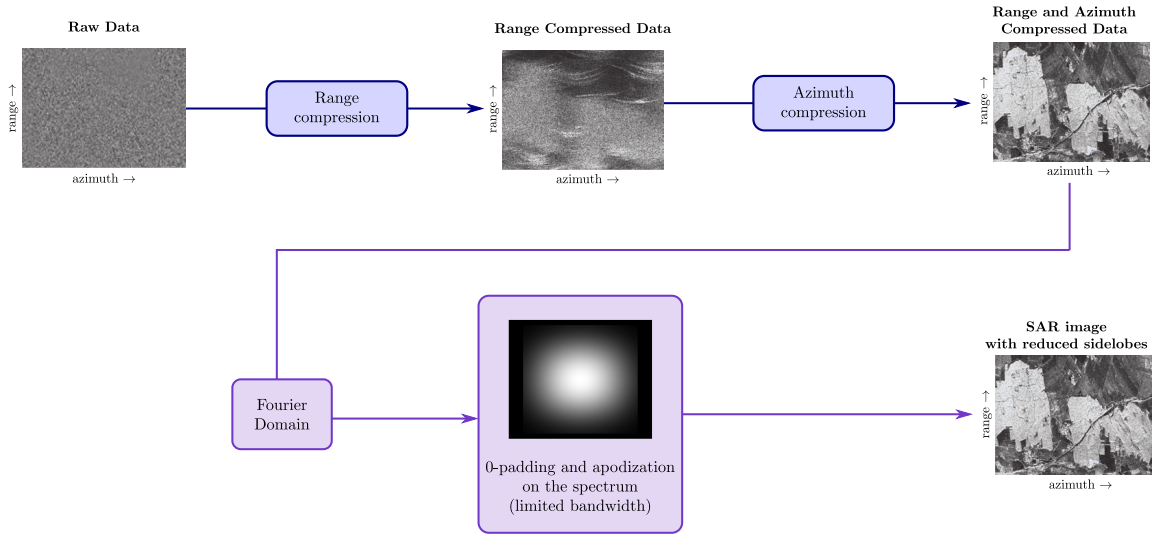


Figure 1.3: Formation of SAR images. The original acquisition is given on the left side. A focusing is done is range, and then in azimuth. The points are localized as the resolution of the image is improved. To reduce the sidelobes of the scatterers, a 0-padding and an apodization function is applied in Fourier domain. After this operation, the speckle is spatially correlated. The figure is inspired by the works [1, 2].

available.

The type of product is chosen based on the application. Different products lead to different levels of noise. This thesis will focus on removing the speckle from SAR images (SLC and GRD images) in various frameworks and applications, all including the use of temporal or polarimetric information to improve the despeckled estimated image.

1.1.3 Examples of applications in remote-sensing

Numerous applications of both active and passive sensors have been developed within the community to tackle environmental issues and challenges.
For example, Sentinel-2 images have been used in [4] to detect and monitor wildfires near the Vesuvio in Italy in July 2018. As shown in Figure 1.6, RGB images contain mainly smoke which is useful to help the local population to know when to evacuate. However, it is difficult to identify the fire pits. When using longer wavelengths (infrared light), the thermal activity can be studied and the pits localized.

Sentinel-1 images can be used to monitor deforestation in equatorial areas on the globe, perform tomography to reconstruct an area in 3D. Being invisible for optical sensors does not mean invisible for SAR sensors: if there is a change in materials or density, it can be detected and tracked with SAR sensors. An exemple is given for oil spill: using SAR images enables tracking of the spill through time and eases the task of local authorities as shown in Figure 1.7 for the Grande America vessel oil spill of March, 12th 2019.

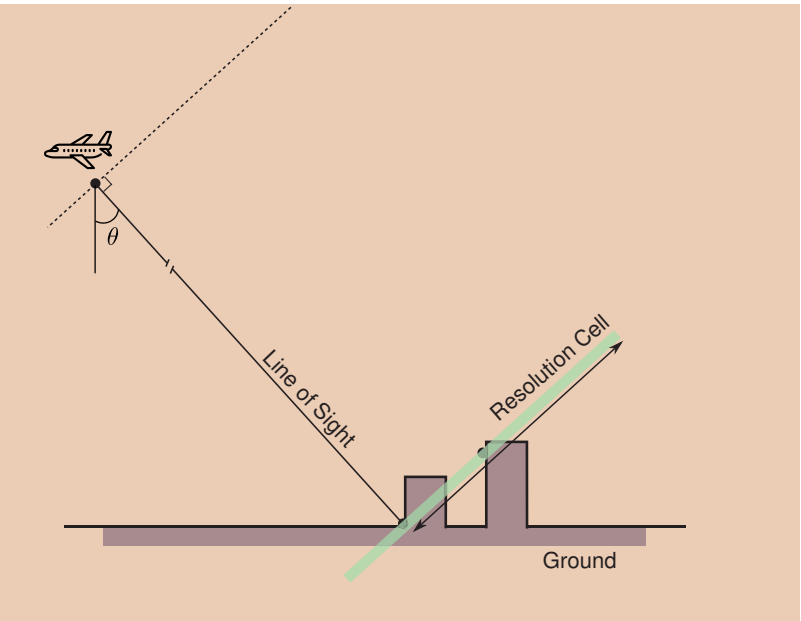


Figure 1.4: Acquisition geometry of SAR images. The sensor, here illustrated by the airplane, is side looking the area of interest with an incident angle θ approximately equal to 30 degrees. Because of the orientation of the line of sight, echoes of scatterers along the axis orthogonal to the line of sight are mixed within one resolution cell represented by the green rectangle. Thus, the responses of the highest point of the right building, and the lowest point of the left building are mixed together. This leads to the flattening effect of elevated objects that can be seen in Figure 1.5. Figure taken from [3].

Because of their numerous advantages, the various information sources within SAR images and the challenges linked to their interpretation, we have only worked with SAR images in this thesis.

1.2 The IMAGES team at Télécom Paris: previous work and outstanding problems

To properly introduce the context of this work, we remind the previous work of the team working in Remote-Sensing in the lab.

The work is at the intersection of image processing and machine learning. All the developed approaches take into account the physics of the acquisition of SAR images.

The focus have been numerous through the years: information extraction (edge and line detectors [5, 6], target extraction [7], lake and narrow river detection [8, 9]); 3D reconstruction in urban areas with SAR tomography ([10, 3, 11]); speckle reduction ([12, 13, 14, 15, 16, 17, 18, 19]).

In the following, we highlight the contribution of the team in despeckling for SAR images to better position our own work.

The problem of despeckling has been tackled by the team with different points of view through

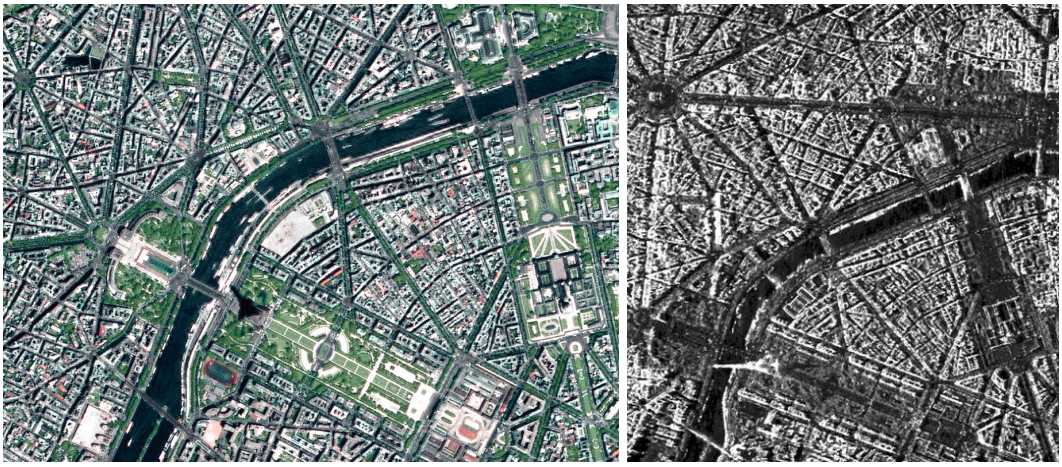


Figure 1.5: Optical and SAR image of Bir Hakeim, Paris. The SAR sensor is on the left side of the image. We can see that the building are *flatten* on the ground, this is particularly visible with the Eiffel tower. These geometrical deformations are due to side looking: the top of the buildings are seen at the same time as the ground because of an incident angle around 30 degrees. Thus, the scattered signal of the top of the buildings and the ground are mixed in one pixel. This is repeated for the windows of the building, starting from the top to the bottom. Optical image: SPOT5 ©CNES; SAR image: TerraSAR-X ©DLR.

the years, all taking into account the physics of the SAR acquisition.

The exploitation of the patch similarity for SAR images has been explored starting with the non-local paradigm developed in [12]. An interferometric version NL-InSAR has also been proposed in [13] and generalized by the NL-SAR algorithm [20].

Because the statistics of the speckle are different from the traditional noise of natural images and its statistics differ from the white Gaussian noise mostly used within the image processing community, the application of traditional denoising algorithms is not easy. The framework MuLoG [14] focused on how to apply a gaussian denoiser to SAR images with possibly multiple channels in order to remove speckle.

Deep learning methods have also been developed, starting with convolutional neural networks pre-trained on additive white Gaussian noise [17]. Even if the deep learning methods have overpowered the traditional ones, the issue of the spatial correlation of the speckle was unresolved: a downsampling step was still needed before despeckling to reduce these correlations. The semi-supervised despeckling algorithm [18] was proposed to tackle this issue. This framework is based on the Noise2Noise work by Lehtinen [21]: the training is supervised by another noisy sample of the same area instead of a groundtruth image. To have pairs of noisy images, changes need to be compensated. Further details are given in Section 4.1. To go even further, the MERLIN framework [19] uses only one single image during the training: the pairs of images are formed using the real and imaginary parts of the Single Look Complex images. The method is explained in section 5.3. Both networks are trained using a negative log-likelihood derived from the statistics of the speckle.

Multi-temporal despeckling has also been studied by the team. The non-local method 2SPPB [15], based on non-local means, exploits the information redundancy in a multi-temporal stack

Figure 1.6: On the left panel, top: aerial view of the wildfires around the Vesuvio, bottom: MODIS (©NASA) image of the area. On the right panel, top: Sentinel-2 RGB image where only the smoke is visible, bottom: Sentinel-2 false color image obtained with Short-Wave InfRared and Near-InfRared bands where we can localize the points of high temperature i.e. the fire pits. Figure extracted from [4].

of images to find similar patches. RABASAR [16] uses a multi-temporal stack of SAR images to compute a super-image and despeckle the ratio image between the noisy and the super-image. Further details on this method will be given in Section 5.2.2.

The work presented in this manuscript is continuing the effort of the team to improve SAR images: the proposed approaches are centered on self-supervised and unsupervised learning for despeckling and on the joint use of several images. We account for the physics of the acquisition in the methods. Even if the fusion of modalities could be relevant in this context, we decided to focus only on SAR images.

A detailed presentation of the contributions of our work is given in chapter 3 after some context (chapter 1) and an introduction on denoising (chapter 2).
Different contexts have been explored: the context of scarce data and the context of abundant data. The context of scarce data will be considered in the application of despeckling for sea ice images where temporal and structural changes are significant because of the ice crackling and shifting rapidly. Temporal stacks are not helpful for the despeckling task and polarimetric information will be used to perform the joint self-supervised despeckling (chapter 4).

The context of abundant data will lead to an approach for multi-temporal despeckling with several images of the same area captured at different times (chapter 5). This will lead to two kinds of approaches developed in this work: the first one based on temporal averaging and the concept of super-image; and the second one using unsupervised learning.

The lack of groundtruth images makes the evaluation of the despeckling methods very difficult. We propose in the last chapter to explore the uncertainty estimation for our models in order to provide a confidence degree to our predictions.

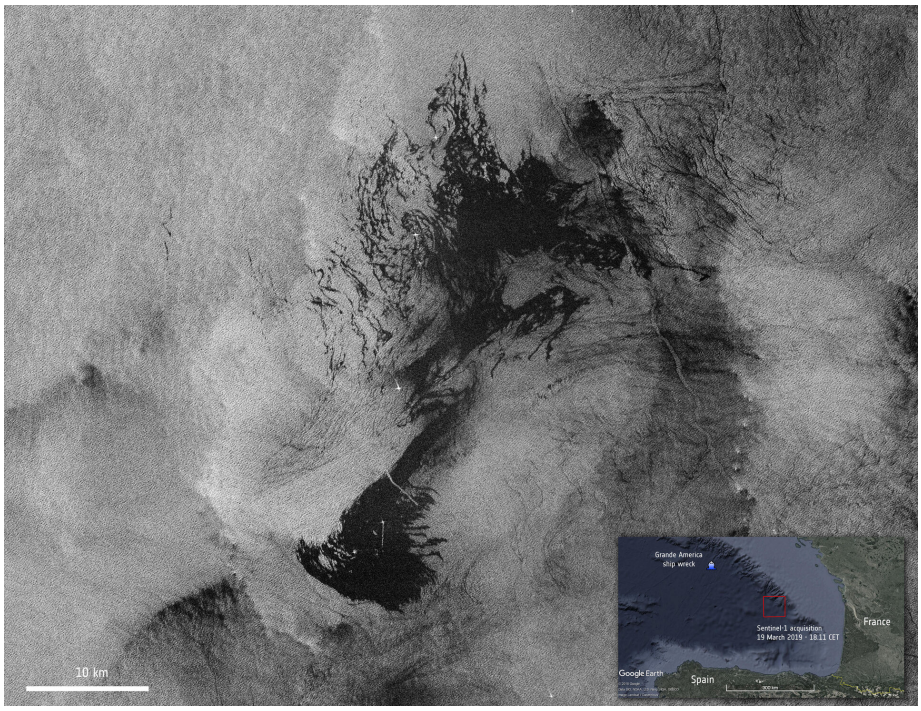


Figure 1.7: Sentinel-1 image of the Grande America oil spill, March 12th 2019. The Italian container ship caught fire near the French West coast and sank hours after. The ship was carrying 2 200 tons of fuel that were spread in the Atlantic ocean. Here we can see the oil spill in black stretching on more than 50km. Marine vessels sent for help are also visible (bright targets). ©ESA official website

Chapter 2

Introduction: from image denoising to despeckling

In this chapter, we introduce in section 2.1 the denoising problem and the usual assumptions on the noise in the literature. The despeckling problem is then developed in section 2.2: the origin and statistics of the speckle are detailed, and a literature review describes the classical and deep learning despeckling techniques.

2.1 Introduction to image denoising: usual assumptions on the noise and literature overview

In this first section, we present a short introduction to the denoising of natural images corrupted by an additive white Gaussian noise. These concepts form an important basis for the development of SAR image despeckling techniques.

2.1.1 Usual assumptions in image denoising

We start by the general expression used in inverse problems in image processing:

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}) + \boldsymbol{n} \tag{2.1}$$

where $\boldsymbol{y}$ is the degraded observation, $\boldsymbol{x}$ is the unknown clean image to be recovered, $\mathcal{A}$ is a degradation operator that is often linear and $\boldsymbol{n}$ is a random realization of a noise term following a given noise distribution.
The operator $\mathcal{A}$ can be any degradation operator such as a blurring operator, an inpainting operator, a projection or a geometric transform. In the image denoising problem, the operator $\mathcal{A}$ is the identity.

The additive model of the noise in equation 2.1, where the distribution of $\boldsymbol{n}$ does not depend on $\boldsymbol{x}$, is a standard assumption in many algorithms proposed in the literature. Furthermore, the noise is generally supposed white and Gaussian as it triggers mathematical properties that are at

the core of various methods. This leads to the following problem

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n} \quad \text{with } \boldsymbol{n} \sim \mathcal{N}\left(0, \sigma^2 \mathbf{I}\right) \tag{2.2}$$

In this simple model, the variance of the noise σ^2 is associated with the level of the noise and does not depend on the signal.

2.1.2 Literature overview

Many methods in image processing have been proposed to tackle image denoising throughout the years. We provide a quick overview of denoising techniques. We distinguish *traditional denoising methods* and *deep learning based methods*.

Traditional denoising methods

Traditional denoising methods can be classified into different types of methods. In this section, we will distinguish the following ones

- Spatial filtering methods
- Variational methods
- Non-local filtering methods
- Transform-domain methods

The first methods to tackle the denoising of natural images corrupted by an additive white Gaussian noise are spatial-domain filter methods, such as the simple mean filtering where the value of each pixel in the image is replaced by the average value of its neighbors within a window with a fixed shape.

The variational methods use image priors and minimize an energy function to produce the denoised estimation. The energy is generally decomposed into two terms. The first term is known as the likelihood/data fidelity term, and the second term is the image prior/regularization term. The likelihood term can be derived by assuming that the observations are distributed according to a Gaussian distribution. Algorithms have been proposed based on different expressions of the image prior. The total-variation (TV) regularization introduced in 1992 by [22] greatly restores the homogeneous areas of the image, but tends to over-smooth the image. Using a sparse representation of the image within the optimization problem has been explored. Each patch of the image is represented as a linear combination of several patches from a given dictionary. This dictionary can be learned from a dataset, or from the image itself using the K-singular value decomposition (K-SVD) algorithm [23]. The sparse representation can be coupled with the non-linear self similarity property of natural images as in the non-local centralized sparse representation (NCSR) model introduced in [24].

In 2005, the Non-Local-Mean algorithm [25] uses the concept of auto-similarity meaning that for every patch within one image, one can find similar patches within the same image. To estimate the denoised image, a patch centered on each pixel of the image is extracted; and the central pixel is replaced by the weighted average of the central pixels of all the similar patches.

Transformed domain methods rely on the observation that the characteristics of the image and the noise are really different in the transformed domain, and it is thus easier to separate the noise

from the clean image. The most famous transformed domain method is the Principle Component Analysis, and its variation Independent Component Analysis [26]. The wavelet transform enables a decomposition of the image into a scale space representation to better separate the main characteristics of the image. However, all wavelet methods rely on the choice of the wavelet basis. The BM3D method [27] is an extension of the Non-Local Means algorithm. For each pixel of the image, similar patches are stacked into a 3D box using block matching. The 3D block is then transformed into a wavelet domain, where a filtering is done. After an inverse transform, the estimated patches are aggregated to compute the final denoised image.

Deep learning based denoising methods

CNN-based methods have also been proposed to tackle the denoising problem. We can distinguish the ones using supervised training and thus needing ground truth images to compute the Loss during the training phase; and the unsupervised methods requiring only noisy image.

In the supervised learning framework, a prior knowledge of the noiseless associated to each noisy image is needed. The Recursively Branched Deconvolutional Network (RBDN) [28] is a generic image-to-image regressor and can thus be applied to image denoising. Zhang et al. [29] proposed the DnCNN and apply a residual formulation to learn the denoising function. The model is trained with a fixed variance of the noise which makes the generalization hard for the network: a training is needed for every level of noise. To tackle this issue, the work [30] introduces the FFDNet which takes an additional input informing the network of the noise variance at each pixel to help the network even if the variance is not constant across the image.

The work Noise2Noise of Lehtinen et al. [21] has been game changing and a pioneer in unsupervised learning. Indeed, no groundtruth data is needed for the training phase in the framework. Assuming that pairs of noisy images of the same scene with independent and identically distributed noise are available, the network will be able to restore the first noisy image by supervising the training with the second one.
The Bayesian framework of the Noise2Void network [31] takes only one single noisy image as input during the training phase. The network is trained to predict each pixel value using only its neighboring pixels (blind spot method).

More recently, vision transformers have been developed and enable a non-local attention mechanism to build a representation of the image [32]. Diffusion networks [33] have also reached excellent results in regression for image processing and more particularly denoising. Transformers and diffusion based architectures are very large and fastidious to train. In this thesis, we have developed the framework to use these king of models, but haven't explored the influence of the architecture and kept it simple in all the experiments. This choice is developed in the next chapter.

2.2 Application to SAR images: the despeckling problem

The despeckling problem we want to tackle in this thesis has some specifics that differ from the denoising problem with natural images. In this section, we introduce the characteristics and statistics of the speckle.

2.2.1 The speckle: origin and statistics

The speckle originates from the summation of the coherent echoes coming from different objects on the ground, and mixed within one resolution cell. Thus, when an area on the ground is observed by the sensor, the backscattered signals coming from various scatterers are mixed and form constructive or destructive interferences. This phenomenon, known as speckle, is linked to the physics of the acquisition and the observed scene itself. It is then impossible to capture speckle-free images with a SAR system (we can still approximate one that could be used with a simulated noise, this will be described in Chapters 4 and 5).

Because the number of coherent echoes within one resolution cell is high, we can apply the central limit theorem to model the distributions of the real and imaginary parts of a single look complex image $z = a + j\,b$. According to Goodman's model [34], at a given pixel, the real and imaginary part are independent and identically distributed according to a Gaussian distribution centered on zero and with a variance equal to $\frac{r}{2}$, with r the real-valued reflectivity of the scene, meaning the expectation of the intensity.
We can then deduce that the complex amplitude z follows a complex circular Gaussian distribution defined by

$$\begin{aligned}
\mathrm{p}(z) &= \mathrm{p}(a + j\,b) \\
&= \frac{1}{\sqrt{2\pi}\,\sqrt{\frac{r}{2}}} \exp\left(-\frac{a^2}{r}\right) \times \frac{1}{\sqrt{2\pi}\,\sqrt{\frac{r}{2}}} \exp\left(-\frac{b^2}{r}\right) \\
&= \frac{1}{\pi r} \exp\left(-\frac{a^2 + b^2}{r}\right) \\
&= \frac{1}{\pi r} \exp\left(-\frac{|z|^2}{r}\right)
\end{aligned} \tag{2.3}$$

Here, we are modeling the statistics of z and thus of the speckle independently of the SAR system response.

We introduce the intensity image as $i = |z|^2$. The speckle is a multiplicative noise, modeled by Goodman as follows

$$i = r \times s \tag{2.4}$$

where s is the complex speckle component.

To reduce the speckle, an averaging step called multi-looking is often performed, leading to the introduction of the equivalent number of looks L linked to the number of images averaged during multi-looking. The higher L, the more images we have averaged and the lower the level of speckle. The intensity of the speckle component is distributed according to a gamma distribution defined by

$$\mathrm{p}(s) = \frac{L^L}{\Gamma(L)}\, s^{(L-1)} \exp(-L\,s) \tag{2.5}$$

with $\Gamma(.)$ the gamma function. The expression of the variance and expectations values are given by

$$\mathbb{E}[s] = 1$$
$$\mathbb{E}[i] = r$$
$$\mathrm{Var}[s] = \frac{1}{L}$$
$$\mathrm{Var}[i] = \frac{r^2}{L}$$

We can see that the variance of the signal depends on the reflectivity r: the highest the reflectivity, the strongest the noise.

The multiplicative property of the speckle makes it difficult to deal with as it is signal-dependent. A log transform is often applied to stabilize the variance and obtain an additive noise with a constant variance throughout the image. The log-intensity of the speckle is distributed according to a Fisher-Tippett distribution:

$$\mathrm{p}(\log s) = \frac{L^L}{\Gamma(L)} e^L \log s \ \exp(-Ls^2) \tag{2.6}$$

The expressions of the variance and the expectation value of the resulting noise are given by

$$\mathbb{E}[\log s] = \psi(L) - \log L$$
$$\mathrm{Var}[\log s] = \phi(1, L)$$

where $\phi(1, L)$ is the polygamma function of order L introduced in the book Abramowitz and Stegun [35], and $\psi(L)$ is the digamma function.

The SAR image z is then processed by the SAR system which has a limited bandwidth. This will introduce spatial correlations of the speckle. These correlations are hard to deal with and numerous methods need a pre-processing step consisting in down-sampling the image by a factor two to reduce the spatial correlation of the speckle. This leads to a loss of resolution. Deep learning methods have been able to deal with spatial correlation when trained on real images [18, 19].

In this chapter, we only present single polarization data. The context of dual-polarization acquisitions will be the subject of chapter 4.

2.2.2 Classical despeckling techniques

In the case of single channel images (amplitude data), despeckling techniques have often been inspired by denoising methods for natural images. As the speckle's statistics and properties are different from the traditional additive white Gaussian noise, the adaptation of conventional methods can be challenging. Thus, spatial filtering was used at the very beginning by applying a mean filter on SAR images. The conclusions have been the same as the ones with additive white Gaussian noise: the despeckled images are blurred. Lee's filter [36] introduced a trade-off between the mean filtered image and the noisy image based on the value of the local variation coefficient.

Non-local approaches based on the search of similar patches have also been developed [37, 38, 12, 39, 40, 13, 41, 42, 43, 44], and rely on the same principle as Non-Local Means [45] with an adaptation to multiplicative and non-Gaussian noise. These techniques build non local neighborhoods that are defined on the basis of pixel similarity and success in restoring thin structures and discontinuity within SAR images.

Variational methods have also been used with various regularization terms such as the Total-Variation [46, 47, 48]. An analysis of similar patches can also be introduced while minimizing the energy (data fidelity term + regularization term) as proposed in [49].

Transformed domain methods were explored through wavelet decomposition [50] and curvelet decomposition [51].

Nevertheless, the adaptation of methods inspired by natural image denoising can be tedious. The MuLoG algorithm [14] circumvents theses adaptations by using a plugin ADMM approach: the restoration of the image alternates between non-linear steps to take into account the statistics of the speckle, and Gaussian denoising steps performed by any Gaussian denoiser, including deep learning based denoiser.

2.3 Despeckling using deep learning based methods

Deep learning algorithms have proved to be very effective in image denoising, and they can deal with non-Gaussian corruptions of images. The training step is time-consuming and requires a relatively large data set. As the speckle is inherent to the scene, ground truth images can not be obtained and the training of a network for despeckling has to deal with this lack of clean data. The inference step on testing data is very fast, which is a very interesting point for real-time applications.

Supervised deep learning techniques were the first to be developed, the pioneering work beeing the framework of Chierchia [52]. To train their network, ground-truth images are computed using temporal stacks of SAR images: the areas that do not change through time are kept. The loss is evaluated using a couple of one speckled image and one temporal average. The network architecture, is inspired from the DnCNN [29].

The method NL-CNN introduced in [53] combines a patch-based non-local filtering and deep learning. The training set is computed using a similar strategy as one used in [54] which can be sub-optimal because of temporal changes.

An alternative is to simulate speckle noise on a ground truth image obtained by temporal averaging [55]. The Multi-Objective CNN-Based Algorithm MONet approach [56] uses a multi-objective Loss function and focuses on spatial details, speckle statistical properties, and strong scatterers identification. Large datasets can be created by adding simulated speckle noise to natural images [57, 58, 59]. However, the statistical distribution of natural images is different from the one of SAR images leading to poor restoration of bright scaterrers. As illustrated in [60], because of spatial correlation of the speckle, results suffer from artifacts.

Self-supervised learning was introduced based on the work Noise2Noise by [21]. Training on actual SAR images has been possible using pairs of images [18, 61] or even single images [19]. Temporal stacks can be used to compute speckled pairs of SAR images. Adversarial learning has been used to produce speckle-free images in [62]. An adaptation of the Noise2Void framework [31] has been proposed by [63], however, the correlation of the speckle can not be taken into account with the Bayesian model used in the blind-spot method and a sub-sampling step is required.

Finally, SLC images are complex-valued: the adaptation of neural networks to complex values is challenging. The works ϕ-Net proposed in [64] is an adaptation of the U-Net to denoise the interferogram of SAR images. In the framework, the real part and the imaginary parts of the complex interferogram are decorrelated, and then fed to the network.

A summary of the deep learning despeckling methods is given in Figure 2.1.

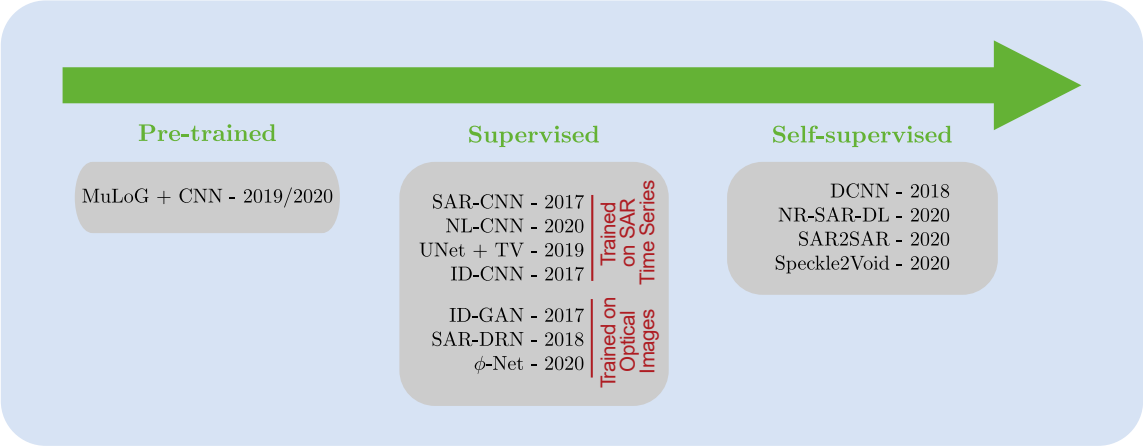
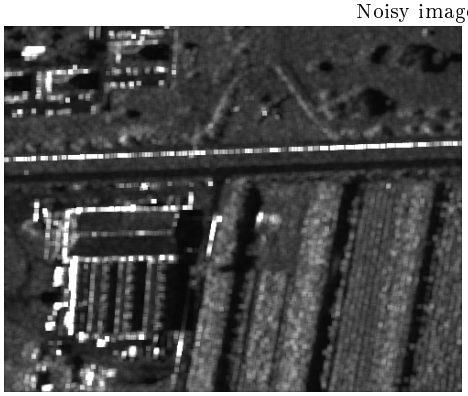


Figure 2.1: Deep learning despeckling methods. Illustration from [65].

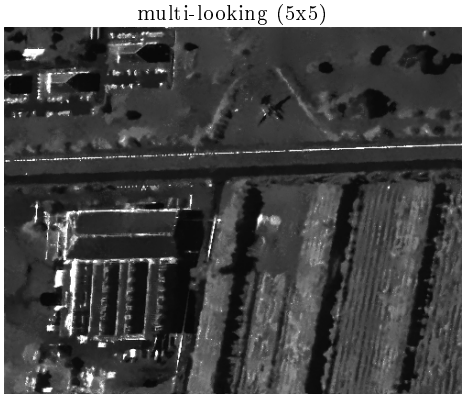
Despeckled images using various methods for SAR despeckling are presented in Figure 2.2. In the following, we will focus on self-supervised methods able to tackle spatial correlation of the speckle [18, 19].

Noisy image, ©ONERA

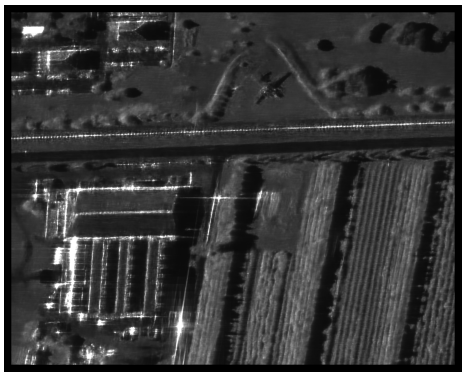


Denoised spatial multi-looking (5x5)

Denoised TV regularization



Denoised NL-SAR



Denoised with deep learning method MERLIN

Figure 2.2: Despeckling of SAR images with various methods illustrated on an image acquired by the SETHI sensor of ONERA.

Chapter 3

Issues addressed in the thesis

This chapter discusses the issues we want to tackle while pointing the possible limits we will face. This Ph.D. work tries to answer to the following question: how can we perform joint despeckling using deep learning approaches and self-supervised/unsupervised learning? Joint despeckling implies the collaborative processing of several images as input, in particular in this thesis: a multi-temporal stack or polarimetric data. The information redundancy within the input data can potentially be exploited to retrieve thin structures such as tiny roads and rivers.

Choices concerning the architecture of the neural networks at the core of the proposed approaches are discussed in following paragraphs. A schematic summary of the contributions of this thesis is given at the end of the chapter.

3.1 Joint despeckling

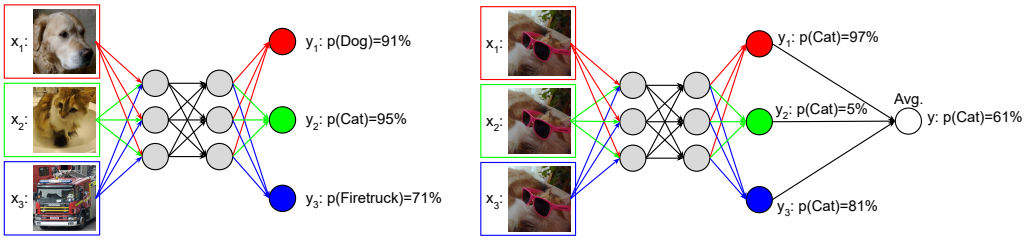
The baseline for our comparisons will correspond to the single image SAR despeckling: this framework takes a single image as input and estimates its reflectivity. This kind of methods can be labeled as Single-Input Single-Output (SISO) methods.

The extension to several images of the same area (acquired at different times or coming from different polarizations) can be achieved using two distinct strategies:

- Multi-Input Multi-Output (MIMO) despeckling: multiple images are fed to the despeckling network, and an equal number of reflectivity images are estimated simultaneously.
- Multi-Input Single-Output (MISO) despeckling: multiple images are fed to the despeckling network, but only the reflectivity of the reference image is predicted.

For various applications and especially real-time applications, the MIMO strategy is far more interesting: one only needs one inference step to despeckle an entire stack of SAR images. However, the regression problem including the prediction of several despeckled images is more difficult to solve.
The motivation of using MIMO strategies comes from [66] where they aim at quantifying uncertainty by using an approach inspired by ensemble methods. They prove that, using a MIMO configuration, we can utilize a single model's capacity to independently train several subnetworks for a specific task. During the inference, the predictions of the independent subnetworks are averaged and the

model is thus more robust without any additional computational cost. The authors empirically show that 3 to 4 subnetworks can be used for a classification task. The framework, illustrated in Figure 3.1 is taking 3 to 4 images as input and performs the 3-4 predictions in parallel.



(a) Training step (b) Testing step

Figure 3.1: Training and testing steps for uncertainty quantification using a MIMO framework and independent subnetworks for classification. Only the first and last layers of the network are changed. (a) Training step: 3 input images are fed to the network corresponding to 3 different classes. The predictions are done on each member thanks to independent subnetworks. (b) Testing step: the same image is fed three times, leading to 3 different predictions from each subnetwork. The final estimation is computed by averaging the 3 predictions. Illustration from [66].

Coming back to our despeckling problem: regression problems are known to be more difficult to solve than classification problems. Proving the independence of different subnetworks is also difficult in our case. It is reasonable to say that 3 or less subnetworks can be used with a MIMO strategy. In the following work, we will apply the MIMO strategy with no more than 2 input images, and we will use the MISO strategy for 3 images and more.
We provide experimental results to show that, on the considered experiment, the performance of the MIMO formulation with 2 input images is comparable to that of the MISO formulation. The framework of the experiment is as follows:

- The experiment has been conducted on a despeckled Sentinel-1 temporal stack produced by the ratio-based multi-temporal despeckling method introduced in [67]. The stack is composed of 25 images of 1024×3072 pixels, representing the city Lelystad in the Netherlands. The speckle is simulated independently on these images during the training.

- Two intensity SAR images of the same area acquired at different times are used for testing. We will note them $\boldsymbol{i}_1$ and $\boldsymbol{i}_2$. These images are generated based on $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ the corresponding groundtruth images. Speckle is independently simulated on both input images with an equivalent number of look L equals to 1 in both cases.

- For the MIMO strategy, the outputs of the network are the estimated reflectivities of both input images $\boldsymbol{i}_1$ and $\boldsymbol{i}_2$. We will note them $\{\tilde{\boldsymbol{r}}_{1,\text{MIMO}}, \tilde{\boldsymbol{r}}_{2,\text{MIMO}}\} = f_{\theta,\text{MIMO}}(\boldsymbol{i}_1, \boldsymbol{i}_2)$ where $f_{\theta,\text{MIMO}}$ is the network function and θ its parameters. During inference, one can directly estimate both denoised images with a single forward pass.

- For the MISO strategy, the output of the network is only the estimation of the reflectivities of the first input image such that $\tilde{\boldsymbol{r}}_{1,\text{MISO}} = f_{\theta,\text{MISO}}(\boldsymbol{i}_1, \boldsymbol{i}_2)$ and $\tilde{\boldsymbol{r}}_{2,\text{MISO}} = f_{\theta,\text{MISO}}(\boldsymbol{i}_2, \boldsymbol{i}_1)$ where $f_{\theta,\text{MISO}}$ is the network function and θ its parameters. The second input image helps the

network to denoise the first one. During the inference phase, one needs to apply the network twice to estimate both $\tilde{\boldsymbol{r}}_{1,\text{MISO}}$ and $\tilde{\boldsymbol{r}}_{2,\text{MISO}}$.

- The quality of the despeckled images is poor in this experiment because of the reduced size of the training set. The main goal is to show that for 2 input images, the MISO and MIMO approaches produce comparable results.

The network is trained in a supervised framework and we define the MIMO and MISO losses used in the training as follows:

$$\mathcal{L}_{\text{MIMO}} = \frac{1}{2}\left(||\log \tilde{\boldsymbol{r}}_{1,\text{MIMO}} - \log \boldsymbol{r}_1||^2 + ||\log \tilde{\boldsymbol{r}}_{2,\text{MIMO}} - \log \boldsymbol{r}_2||^2\right) \tag{3.1}$$

$$\mathcal{L}_{\text{MISO}} = ||\log \tilde{\boldsymbol{r}}_{1,\text{MISO}} - \log \boldsymbol{r}_1||^2 \tag{3.2}$$

The results given in Figure 3.1 show minor differences between the MISO and MIMO strategies. By looking in details, one could eventually see that with the MISO approach, the estimated images are less blurred.

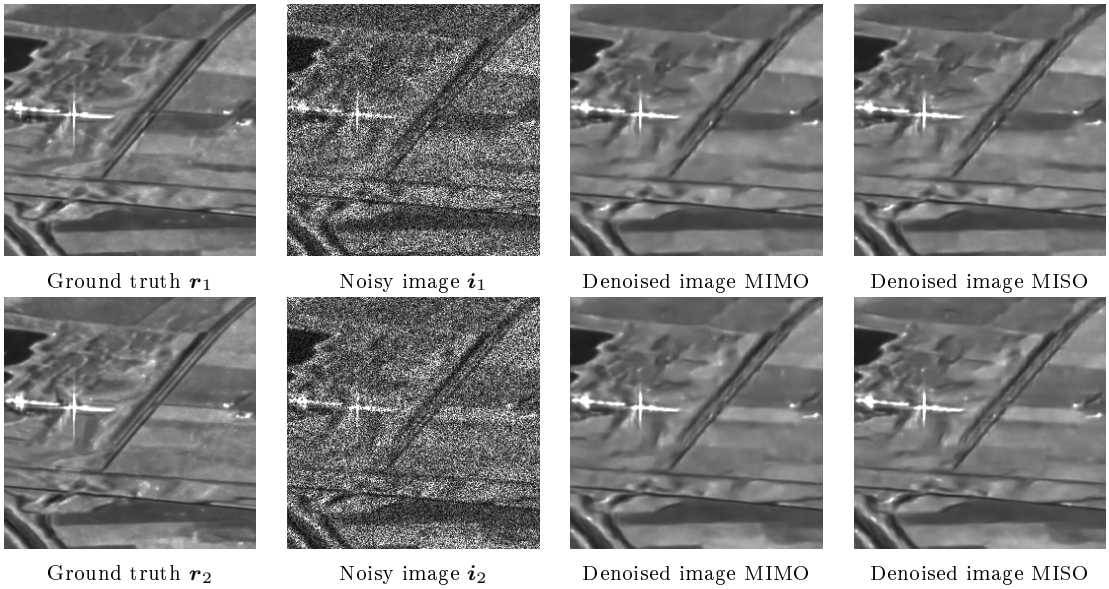


Table 3.1: Comparison of the MIMO and MISO strategies for 2 input images. During the inference, the despeckled images are estimated at once with the MIMO strategy whereas 2 forward passes are needed with the MISO strategy.

The MIMO strategy is computationally more efficient once the network is trained but limited in its application to a small number of input images. In this thesis, we have worked with data where only 2 input images are available at each time and temporal stacks were not available (see Chapter 4): in this case, we will stick to the MIMO strategy. We have also worked with multi-temporal stacks of urban areas SAR images where a higher number of images can be fed to the network at once (see Chapter 5): in this case, the MISO strategy is more relevant.

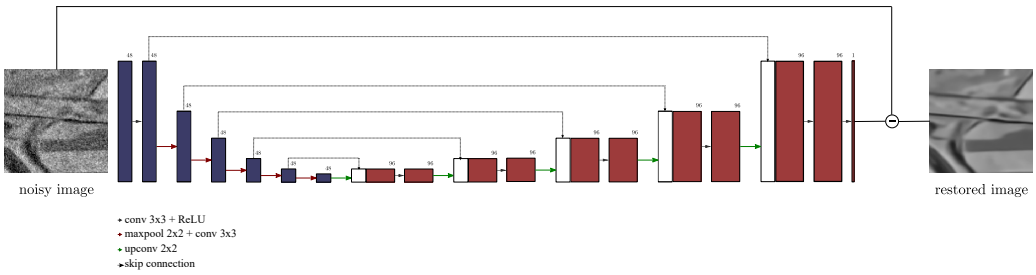


Figure 3.2: Schema of the UNet inspired architecture of the network used in the thesis. The input and output layers change depending on the application and the chosen strategy: MIMO in chapter 4, and MISO in chapter 5. The training is residual, meaning that the network is predicting the speckle in the noisy image. A normalization is applied to input images and consist in a log-transform and a normalization of the input in the range $[0, 1]$. More details on this operation are given in Annex A.

3.2 Focus on training strategies

Even if the ability of the network to deal with changes between the input channels is linked to its capacity, the architectures of the networks proposed in this work remain quite simple. We do not focus on finding the optimal architecture for one specific task and leave this problem of architecture optimization to further works.
The network architecture used throughout this thesis is a UNet network introduced in [18, 19] and detailed in Figure 3.2. This architecture has proven to be very effective and easily trainable in a reasonable time. Changes are made on the first and last layers when the number of input and output images increases.
The Loss functions optimized during the training phases are always derived from the statistics of speckle in SAR images to take the physics of the acquisition into account.

3.3 Structure of the manuscript

Three chapters presenting the thesis contributions follow.

The first one (chapter 4) describes our proposed joint polarimetric despeckling for sea ice images. As structural changes happen very quickly on sea ice images because of the fast shifting of ice on the sea, it is difficult to exploit multi-temporal stacks of sea ice images to improve despeckling. Our approach uses the polarimetric information (HH and HV channels) of GRDM Extra Wide Sentinel-1 images in a MIMO framework.

The second chapter (chapter 5) focuses on our contributions in multi-temporal despeckling in a context where numerous images of the same area acquired at different times are available. Even if there are challenges linked to pre-processing (such as registration of the stack of images), it is beneficial to use as many images as possible. Multi-temporal despeckling requires robustness to temporal changes.

Lastly, a third chapter (chapter 6) synthesizes our work and explores the problem of uncertainty estimation. This work is motivated by the lack of universal metrics to assert despeckling results when no groundtruth image is available.

Part II

Contributions

Chapter 4

Joint polarimetric despeckling of GRDM Extra Wide Sentinel-1 data and application to sea ice images

Publications related to this chapter:

- *Despeckling of dual-pol GRD sentinel-1 images in extra-wide mode by deep learning* (Oral presentation), Ines Meraoumia, Debanshu Ratha, Emanuele Dalsasso, Loïc Denis, Florence Tupin, Andrea Marinoni, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference)**, 2023.
- *Joint despeckling and thermal noise compensation, application to Sentinel-1 images of the Arctic*, Ines Meraoumia, Debanshu Ratha, Emanuele Dalsasso, Johannes Lohse, Florence Tupin, Andrea Marinoni, Loïc Denis, **in preparation**, 2023

In this chapter, we will present the work jointly done with the Arctic University of Tromso in Norway founded by the French-Norwegian project COSMIC (Advanced Processing of SAR Images for the Arctic).

For dual-pol sensors, we want to process jointly the polarimetric information (HH and HV images) to improve the despeckling. In this case, the sensor is using one polarization of the wave for the emission (H here) and 2 polarizations H and V for the wave reception. As the main application of this work is the study of sea ice, we will first introduce the Sentinel-1 products used in the chapter and the challenges related to sea ice images in section 4.1.

A dual-polarimetry joint despeckling network is proposed based on the joint restoration of the intensity images in section 4.2.3. This method is inspired by the existing framework SAR2SAR. The network is trained in a self-supervised way using pairs of noisy images during the training, as it has been proposed in the Noise2Noise framework [21].

We are interested in low reflectivity areas such as water and ice where the thermal noise is visible

and has an impact on despeckling. A despeckling network is proposed in section 4.3 to deal with the thermal noise component on sea ice images.

4.1 Context and challenges related to Sentinel-1 GRDM EW sea ice images

SAR images are very suited to observe the Arctic because of its longer dark periods, frequent precipitations and cloud cover. Besides, subzero winter conditions enable freezing of the ocean into sea ice, which must be navigated in the dark. Furthermore, the study of sea ice concentration, its extent, ice type, the melting of northern glaciers and calving of Greenland glacier ice sheet is of interest to assess the burning questions of climate change that is affecting the world.

With the particular aim to monitor sea ice, the European Space Agency (ESA) provides dual polarimetric SAR Ground Range Detected (GRD) products available in medium resolution from its spaceborne Sentinel-1 sensors for TOPSAR acquisitions in Extra Wide (EW) swath mode. GRDM-EW SAR images are multi-looked intensities projected to the ground range using the Earth Ellipsoid model. It provides a ground range coverage of approximately 400 km in five different sub-swaths. The multi-look factor of the first sub-swath is approximately equal to 15, while for the others it is approximately equal to 10. The information provided by ESA on this product is given in Figure 4.1. These images are intensity images and there is no phase information available.

Beam ID	EW1	EW2	EW3	EW4	EW5
Spatial Resolution rg x az (m)	90.9x90.1	93.1x89.4	93.3x86.9	93.8x86.5	95.1x90.1
Pixel spacing rg x az (m)	40x40	40x40	40x40	40x40	40x40
Incidence angle (°)	23.7	30.9	36.2	40.9	44.5
Equivalent Number of Looks (ENL)	15.2	9.7	9.6	9.5	9.6
Radiometric resolution	1.0	1.2	1.2	1.2	1.2
Number of looks (range x azimuth)	6 x 3	6 x 2	6 x 2	6 x 2	6 x 2
Range look bandwidth (MHz)	4.8	3.3	2.8	2.4	2.2
Azimuth look bandwidth (Hz)	88.5	87.9	86.2	85.3	88.2
Range Hamming weighting coefficient	0.60	0.70	0.72	0.75	0.75
Azimuth Hamming weighting coefficient	0.60	0.61	0.62	0.63	0.60

Figure 4.1: Information on Sentinel-1 GRDM EW images. The beam ID refers to the number of sub-swath within the image. The equivalent number of looks L is higher on the first sub-swath (15.2) and does not vary a lot among the others (between 9 and 10). The variance of the speckle is inversely proportional to L, meaning that the level of noise changes in the image. ©ESA.

When working with Sentinel-1 GRDM EW images, the challenges (apart from the lack of reliable validation data) are related to the speckle fluctuations and the thermal noise which adversely affect the signal, in particular in low backscatter regions in the cross polarization channel. The training strategy and the training data set are described in section 4.2

In this chapter, we propose to perform a joint filtering of the HH and HV images using a self-supervised method inspired by the method SAR2SAR [18]. The joint processing of dual-pol images allows the network to exploit the common information on the underlying scene, leading to a better restoration of the reflectivity than an independent restoration of each polarimetric channel.

The dual-polarimetry despeckling network described in section 4.2.3 removes the speckle from intensity images, the method developed in section 4.3 accounts for the thermal noise while despeckling SAR images.

4.2 Training strategy

Sentinel-1 GRDM EW images are intensity images. In the literature, various despeckling networks have been proposed. We developed a method based on the SAR2SAR network [18] because it is a self-supervised learning framework, robust to the spatial correlations of speckle and applicable to intensity images.

4.2.1 Self-supervised training and the SAR2SAR method

Self-supervised strategies provide ways to train a despeckling network in the absence of ground-truth. The SAR2SAR framework [18] requires pairs of SAR images with decorrelated speckle and can be applied both to Single Look images or multi-looked ground-projected data. It is robust to the spatial correlations of speckle. Speckle2Void [63] is predicting the value of one pixel of the image based on its neighborhood, assuming pixel independence. It can be applied to single images but it requires spatially decorrelated speckle. MERLIN [19, 68] exploits the real and imaginary parts of SLC images and is robust to spatial correlations of speckle, it can be applied to unpaired images [19] or multi-temporal stacks [68]. Yet, due to the lack of phase information, it cannot be directly applied to GRDM data. SAR2SAR is therefore the most adapted framework to develop a technique for Sentinel-1 GRDM images acquired in extra-wide mode.

SAR2SAR training is composed of three phases: in phase A, the network is pre-trained on images corrupted by synthetic speckle; in phases B and C, the network is fine-tuned on real SAR images, learning sensor-specific features (e.g. spatial correlation, content, resolution). A schema of all the phases of the training is given in Figure 4.2.

The loss function used in [18] corresponds to the cumulative negative log-likelihood derived from Goodman's speckle model (where the log speckle component is distributed according to a Fisher-Tippett distribution) over all the pixels:

$$\begin{aligned}\mathcal{L}_{\text{SAR2SAR}}\left(f_\theta(\boldsymbol{y}_1), \boldsymbol{y}_2\right) &= -\log \mathrm{p}(\boldsymbol{y}_2 | f_\theta(\boldsymbol{y}_1)) \\ &\approx \sum_k f_\theta(y_{1,k}) - y_{2,k} + \exp(y_{2,k} - f_\theta(y_{1,k}))\end{aligned} \tag{4.1}$$

where the approximation is verified when spatial correlations of speckle are neglected; f_θ is the network with parameters θ; $(\boldsymbol{y}_1, \boldsymbol{y}_2)$ is the pair of noisy log-intensity images, the first one is fed to the network and despeckled while the second one is used as target for supervision.

In our extension to dualpol despeckling, we define the loss function as the sum of the co-polar and cross-polar restoration losses. We also suppose in the following that the polarimetric coherence

has a limited impact on the result i.e. the HH and HV channels are not correlated. The loss function is thus written as:

$$\mathcal{L}_{\text{SAR2SAR}}(f_\theta(\boldsymbol{y}_{1,\text{HH}}), \boldsymbol{y}_{2,\text{HH}}, f_\theta(\boldsymbol{y}_{1,\text{HV}}), \boldsymbol{y}_{2,\text{HV}}) = \mathcal{L}_{\text{SAR2SAR}}(f_\theta(\boldsymbol{y}_{1,\text{HH}}), \boldsymbol{y}_{2,\text{HH}}) + \mathcal{L}_{\text{SAR2SAR}}(f_\theta(\boldsymbol{y}_{1,\text{HV}}), \boldsymbol{y}_{2,\text{HV}}) \tag{4.2}$$

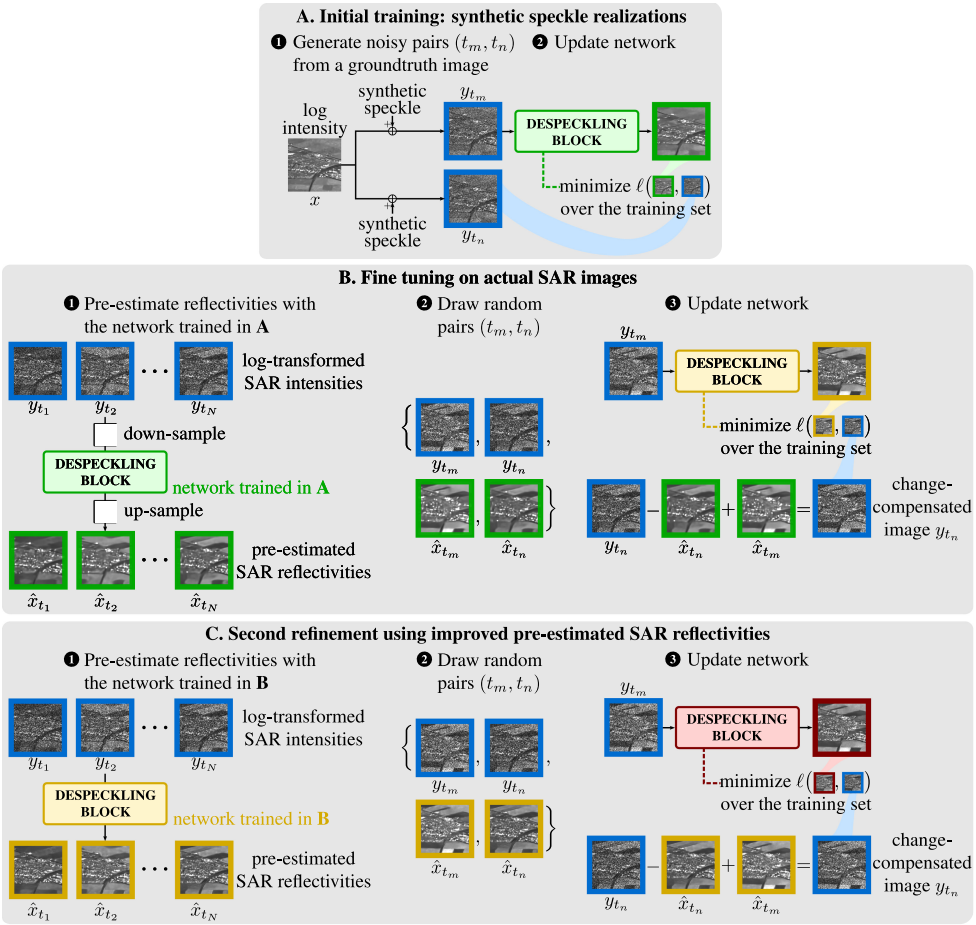


Figure 4.2: Training strategy of the method SAR2SAR [18]. The training phase A is performed on simulated speckle. A pair of noisy samples simulated from the same ground truth image is used for the evaluation of the loss function. Phase B is a fine tuning on actual SAR images. The noisy pair of images is selected within a multi-temporal stack of SAR images. In order to compensate the changes, prior images $\hat{x}_{t_i}$ (with $i \in \{1, \ldots, N\}$) are computed by despeckling the stack with the network obtained in phase A. They are then used to compensate the changes for the target image y_{t_n}. Phase C is a fine-tuning based on the same principle, but the prior images are computed using the network trained in phase B. Illustration extracted from [18].

4.2.2 Building the training data set

The application of the SAR2SAR self-supervised training strategy requires the availability of pairs of images with decorrelated speckle and limited temporal changes. Spatial structures should remain at the same location, yet changes of reflectivity can be handled through the change compensation step.

During the phase A of the training, the speckle is simulated on a RADARSAT-2 stack of 17 images of 679×932 pixels. The ground truth images have been computed using the RABASAR methods [16] where a super-image is constructed by averaging all the images in the stack and the ratio between the noisy image and the super-image is despeckled by the MuLoG algorithm [14]. For each image, the covariance matrix between the HH, HV and VV channels is estimated. To be as close as possible to the case of Sentinel-1 GRDM images, the network is trained with patches corrupted by a simulated speckle with an equivalent number of looks L chosen randomly for each patch in $\{9, 10, 15\}$.

The phase B of the training requires image pairs or multi-temporal stacks of Sentinel-1 GRDM EW images in order to build pairs of images for the evaluation of the loss. The changes are compensated as shown in Figure 4.2, but consequent structural changes are harder to compensate perfectly. Sea ice undergoes too marked changes to be adapted for SAR2SAR training. We preferred selecting areas over land that are more stable.

As the dynamic range of sea ice images is not the same as the one of land images, we need to ensure that our network is robust to these distribution shifts. A discussion on the impact of distribution shift is proposed in Annex A. During the training, we artificially shift the dynamic range of the input images by multiplying the intensities by a factor drawn in the range $[0.1, 1]$ so that the network can also be applied to sea ice images with lower reflectivity values.

We built two multitemporal stacks of Sentinel-1 EW GRD Medium Resolution images (with 8 and 12 images per stack, respectively, each image composed of $10\,000 \times 10\,000$ pixels approximately) near the mouth of the river Ob in north of Russia. The dual-polarization images (HH+HV) in the stacks were observed in summer (August and September) in years 2017, 2018, or 2019, with identical acquisition characteristics and a minimum temporal gap of 12 days. Since land is not covered by snow in these images, there is a significant contrast between the different spatial areas (forests, fields, urban structures, lakes, coastal line, ...). This is beneficial to learn how to restore a wide diversity of spatial structures. Their common footprints are shown in Figure 4.4 represented on the globe. A natural color image of the surrounding region, is provided in Figure 4.3 overlapping the period of acquisition of the selected Sentinel-1 images and the HH polarimetric channel of the first image of the two stacks are shown in Figure 4.5.

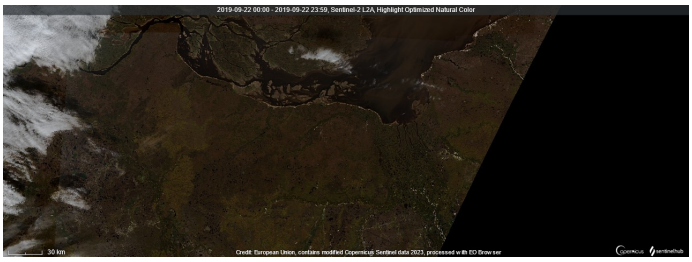


Figure 4.3: Highlighted Optimized Natural color image at the mouth of Ob from Sentinel-2 on 22-09-2019

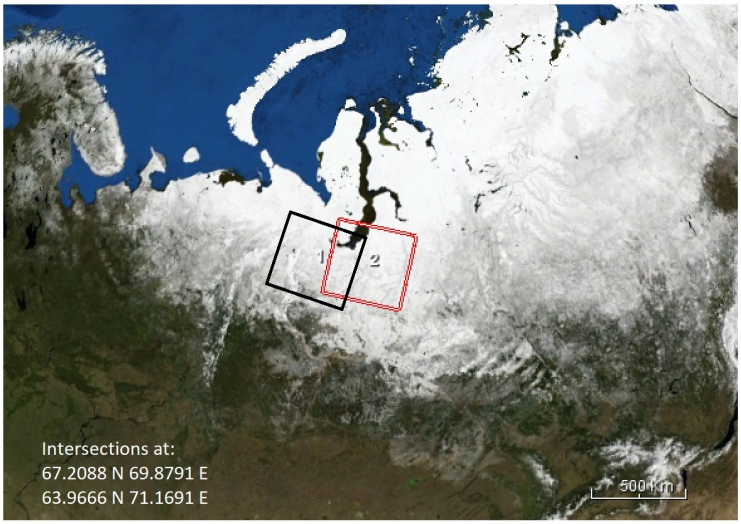


Figure 4.4: The footprints of the two regions selected for training dataset represented on the globe.

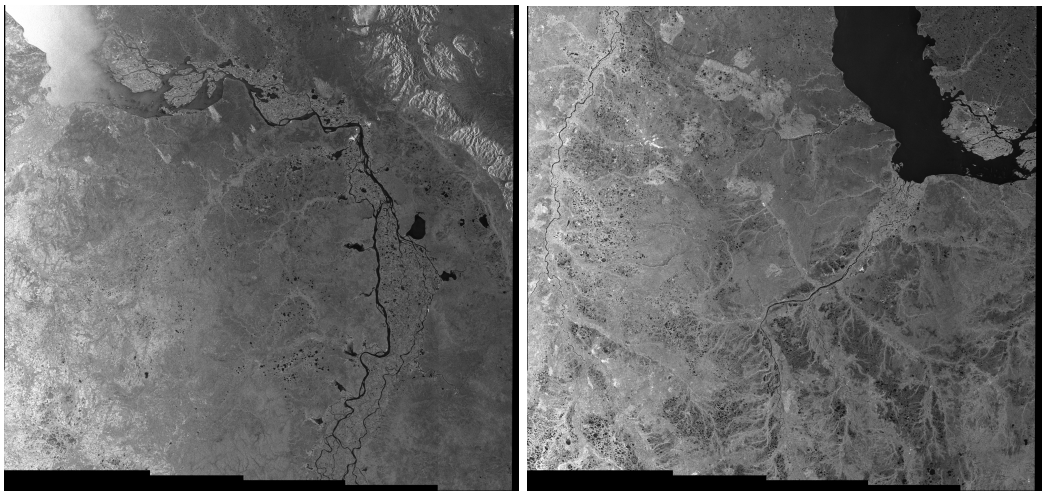


Figure 4.5: Sentinel-1 GRDM EW HH first images of the two multi-temporal stacks used for training (river Ob, north of Russia). Each image contains approximately 10 000 000 pixels.

4.2.3 Dual-polarimetric despeckling network

To perform dual-polarimetric despeckling, we trained a network with the two polarimetric channels as input and the loss function in equation 4.2, so that the network can process jointly the common information and produce better restoration results. The training phases A and B are performed on the RADARSAT-2 and the Sentinel-1 GRDM EW images respectively. The hyperparameters used for the training phases are given in Table 4.1

	Synthetic speckle Phase A	Actual speckle phase B
# **stacks**	1	2
# **images**	17	20
avg images/stack	17	10
patch size	256×256	256×256
batch size	12	12
# **patches**	5304	11 640
# **batches**	442	570
# **epochs**	30	30
learning rate	10^{-3}	10^{-3}
	10^{-4} after 10 epochs	10^{-4} after 10 epochs
	10^{-5} after 20 epochs	10^{-5} after 20 epochs

Table 4.1: Training parameters of dual polarimetric joint despeckling. The network proposed in section 4.2.3 and the network proposed in section 4.3 are trained with the same hyperparameters.

The experimental results are given in Figure 4.6 for the phase A, and Figure 4.7 for the phase B. We can see that the joint despeckling produces better estimation of the reflectivity image, especially for thin structures such as tiny rivers.

As the application of this work is the study of sea ice, the thermal noise can not be neglected. Indeed, the reflectivity values are lower than the ones observed on land, and thus of the same order of magnitude as the thermal noise floor. We need to take the thermal noise component into account for sea ice image despeckling (see Figure 4.8).

4.3 Training the network to account for the thermal noise compensation

The thermal noise is particularly visible in low-reflectivity areas such as water or young smooth ice. The thermal noise component has discontinuities between each of the sub-swath as visible in Figure 4.8. When the reflectivity values are smaller than the noise floor, the degradation can be difficult to invert. Thermal noise appears as a background component with a low reflectivity $\boldsymbol{\sigma}^0_{\text{th}}$ that adds to the reflectivity $\boldsymbol{\sigma}^0$ of the SAR scene. Fluctuations of the intensity in the SAR images are then proportional to $\boldsymbol{\sigma}^0 + \boldsymbol{\sigma}^0_{\text{th}}$. While areas with strong reflectivities $\boldsymbol{\sigma}^0 \gg \boldsymbol{\sigma}^0_{\text{th}}$ are not significantly affected by thermal noise, lower-reflectivity regions suffer from increased fluctuations compared to the level of fluctuations that would occur due to speckle alone. Furthermore, estimations of the reflectivity are biased unless the thermal floor level is removed.

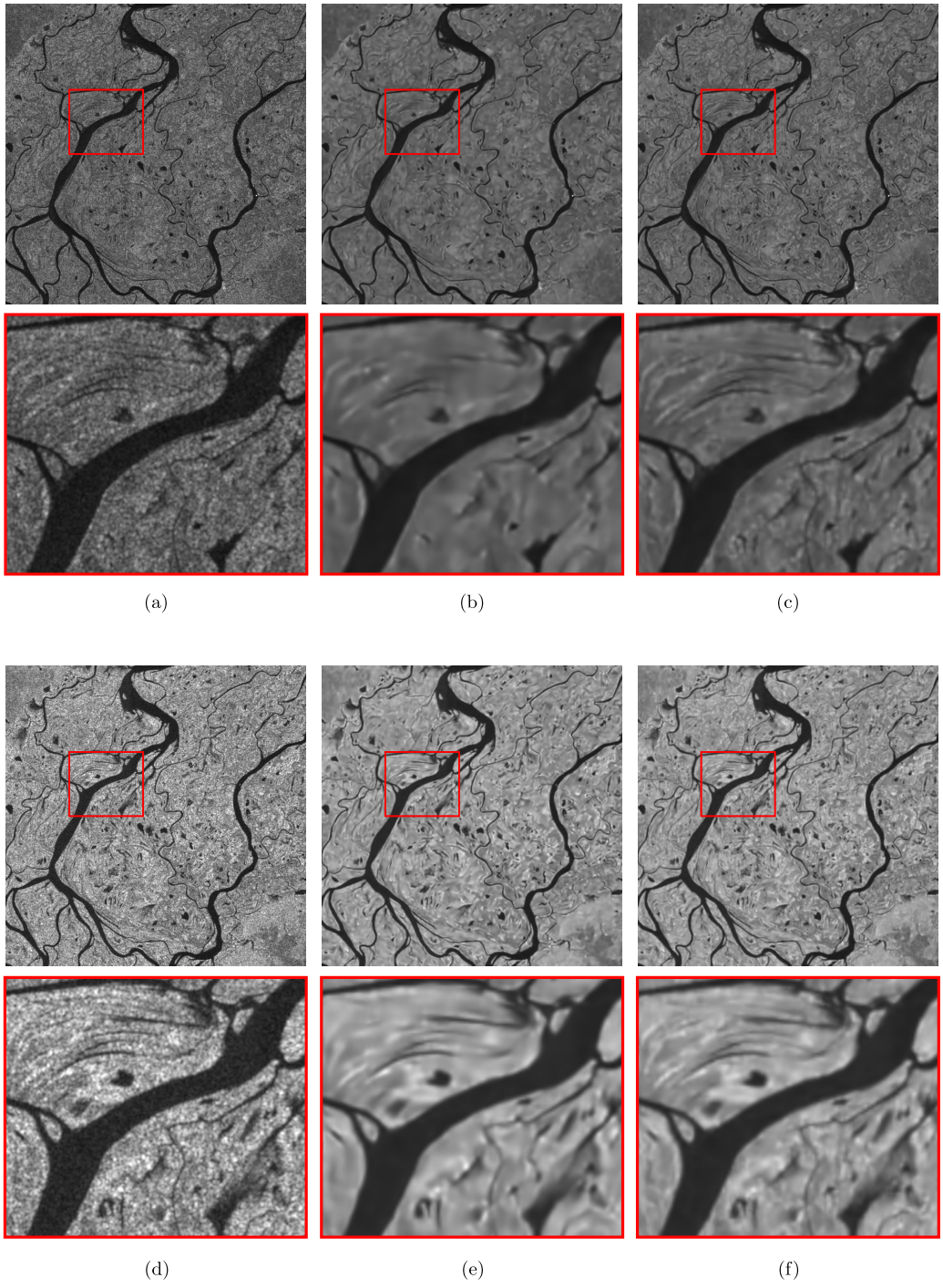


Figure 4.6: Phase A: despeckling results on Sentinel-1 EW GRD Medium Resolution images, river Ob, Russia: the first row shows the HH channel, the second row the HV channel. Images (a) and (d): speckled images; images (b) and (e): restored images with an independent processing of the channels i.e. the SAR2SAR network trained on simulated speckle (phase A); images (e) and (f): dual-polarimetric despeckling network where HH and HV are processed jointly trained on simulated speckle (phase A). As the network is trained on simulated speckle, it can not deal with the spatial correlation of the speckle: a down sampling step is needed before feeding the network with real images. An up sampling step is then performed to match the initial resolution of the images.

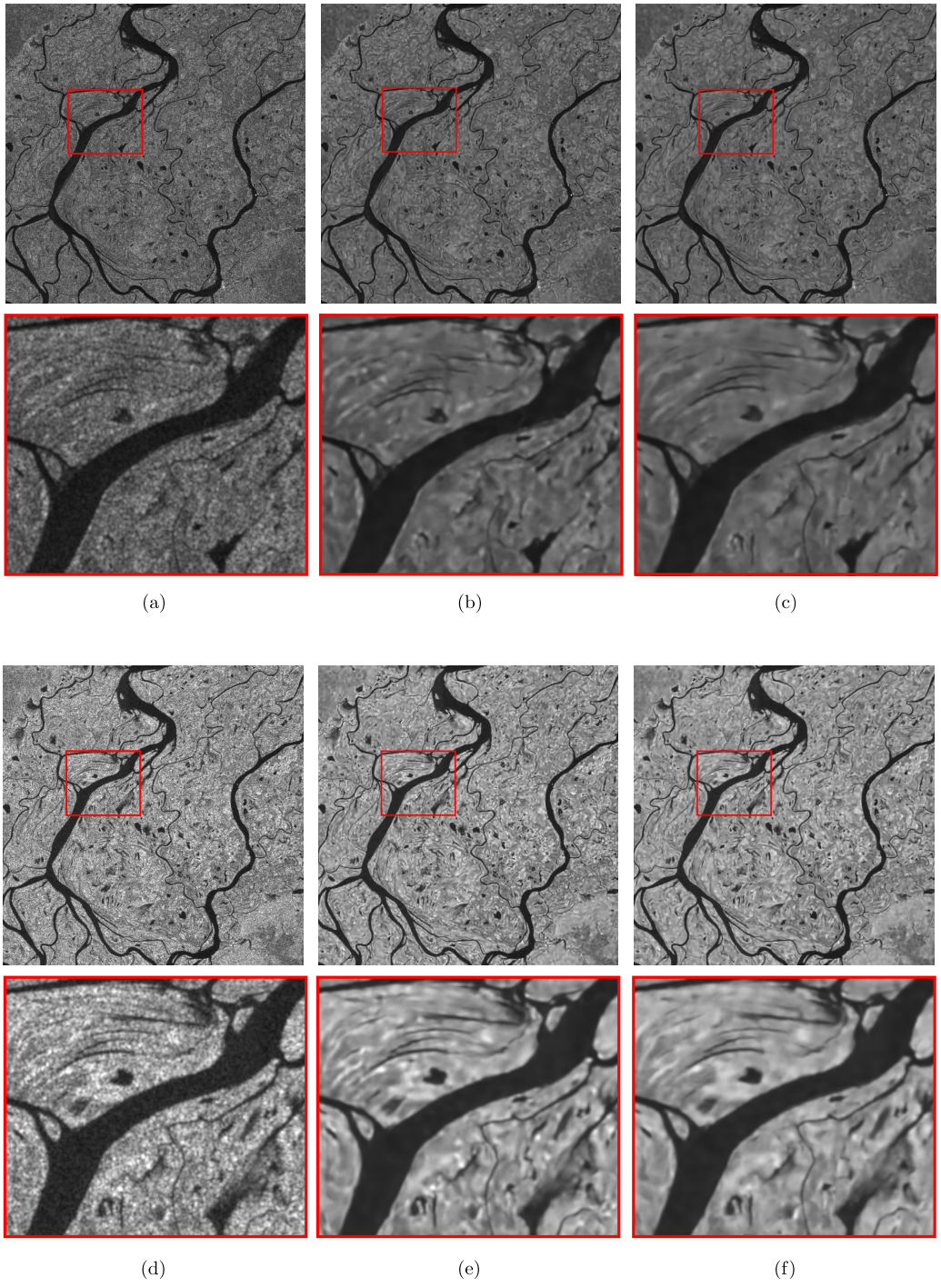
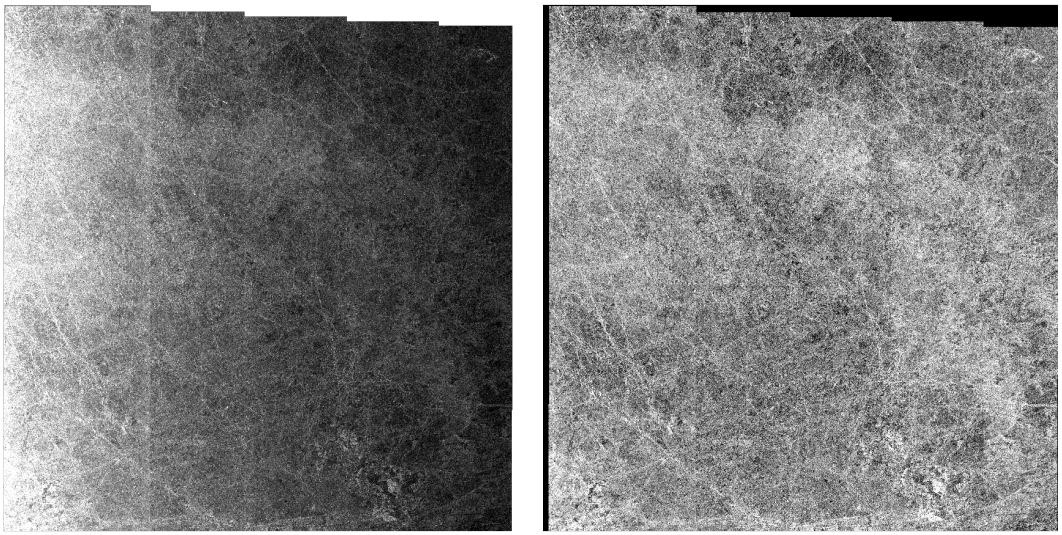
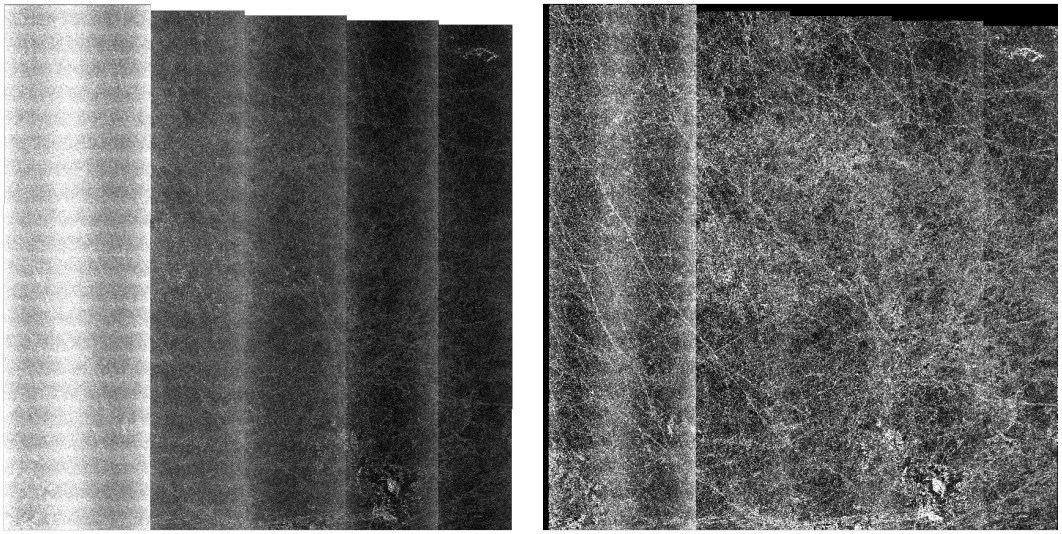


Figure 4.7: Phase B: despeckling results on Sentinel-1 EW GRD Medium Resolution images, river Ob, Russia: the first row shows the HH channel, the second row the HV channel. Images (a) and (d): speckled images; images (b) and (e): restored images with a single polarization processing i.e. the SAR2SAR network trained on the same data set described in 4.2.2, where HH and HV images are considered as two independent samples during the training; images (e) and (f): dual-polarimetric despeckling network where HH and HV are processed jointly. Because in phase B the network is fine tuned with real data, it is robust to speckle spatial correlation and no down-sampling step is needed.



HH noisy image $\boldsymbol{I}_{\mathrm{HH}}$ Corrected image σ^0_{HH}



HV noisy image $\boldsymbol{I}_{\mathrm{HV}}$ Corrected image σ^0_{HV}

Figure 4.8: Sentinel-1 GRDM EW images of sea ice in the Artic. The reflectivity values of the image are lower or equal to the thermal noise floor: thermal noise has a huge impact on the interpretation of the image and is visible, especially on the cross polarization HV. Left column: HH and HV images in amplitude with the thermal noise component. We can see that the antenna gain on the first sub-swath is higher than the rest of the image. The fluctuations of the thermal noise are visible (horizontal strip patterns). Right column: the HH and HV images have been corrected i.e. the thermal noise component has been removed using the Korosov algorithm [69]. This correction is not perfect, but it clearly makes the interpretation of the images easier.

4.3.1 Removing the thermal noise component

For sea ice and oceanographic applications, the calibrated noise vectors provided with GRD data by ESA for NESZ (Noise Equivalent Sigma Zero) subtraction has not been found satisfactory. In a recent work by Korosov et al. [69], a novel correction of thermal noise annotation in the range direction is used to produce an efficient denoising method. This work will serve as the state-of-art for the thermal noise floor removal of the cross polarization channel in S1 GRDM EW mode data for the purpose of our study.

In this chapter, we work with GRD images and the dual-polarizations HH and HV. GRD images are formed with complex amplitude SAR images. Their formation is described in our approximate generative model for dual-polarimetric SAR imaging given in Figure 4.9 (right part of the figure). First we introduce $\boldsymbol{z} \in \mathbb{C}^{2N}$ the vector-image containing the HH and HV complex amplitudes each composed of N pixels. We have

$$\boldsymbol{z} = \begin{pmatrix} \boldsymbol{z}_{\mathrm{HH}} \\ \boldsymbol{z}_{\mathrm{HV}} \end{pmatrix}$$

The polarimetric covariance matrix $\boldsymbol{\Sigma}$ of $\boldsymbol{z}$ is written as

$$\begin{aligned} \boldsymbol{\Sigma} &= \mathbb{E}\left[\boldsymbol{z}\boldsymbol{z}^*\right] \\ &= \mathbb{E}\left[\begin{pmatrix} \boldsymbol{z}_{\mathrm{HH}} \\ \boldsymbol{z}_{\mathrm{HV}} \end{pmatrix} (\boldsymbol{z}^*_{\mathrm{HH}} \ \boldsymbol{z}^*_{\mathrm{HV}})\right] \\ &= \begin{pmatrix} \mathbb{E}[\boldsymbol{z}_{\mathrm{HH}}\, \boldsymbol{z}^*_{\mathrm{HH}}] & \mathbb{E}[\boldsymbol{z}_{\mathrm{HH}}\, \boldsymbol{z}^*_{\mathrm{HV}}] \\ \mathbb{E}[\boldsymbol{z}_{\mathrm{HV}}\, \boldsymbol{z}^*_{\mathrm{HH}}] & \mathbb{E}[\boldsymbol{z}_{\mathrm{HV}}\, \boldsymbol{z}^*_{\mathrm{HV}}] \end{pmatrix} \end{aligned}$$

We know that the relation between the square modulus of $\boldsymbol{z}$ and the intensity image $\boldsymbol{i} = (\boldsymbol{i}_{\mathrm{HH}} \ \boldsymbol{i}_{\mathrm{HV}})^T$ is $|\boldsymbol{z}|^2 = \boldsymbol{i}$. By definition, the reflectivity $\boldsymbol{r}_{\mathrm{HH}}$ and $\boldsymbol{r}_{\mathrm{HV}}$ are defined as

$$\begin{aligned} \boldsymbol{r}_{\mathrm{HH}} &= \mathbb{E}[\boldsymbol{i}_{\mathrm{HH}}] \\ \boldsymbol{r}_{\mathrm{HV}} &= \mathbb{E}[\boldsymbol{i}_{\mathrm{HV}}] \end{aligned}$$

We can then write the covariance matrix $\boldsymbol{\Sigma}_n \in \mathbb{C}^{2\times 2}$ at pixel n by

$$\boldsymbol{\Sigma}_n = \begin{pmatrix} r_{\mathrm{HH},n} & \sqrt{r_{\mathrm{HH},n}\, r_{\mathrm{HV},n}}\, \rho_n\, e^{j\beta_n} \\ \sqrt{r_{\mathrm{HH},n}\, r_{\mathrm{HV},n}}\, \rho_n\, e^{-j\beta_n} & r_{\mathrm{HV},n} \end{pmatrix}$$

where $\rho_n \in [0,1]$ is the polarimetric coherence and $\beta_n \in [-\pi, \pi[$ is the polarimetric phase. In order to match with the notations used in the literature in thermal noise removal, we will denote the polarimetric response of the scene (i.e. reflectivity) by $\boldsymbol{\sigma}^0$ in the following.

The right-hand-side of figure 4.9 illustrates the steps that lead to ground-detected images such as Sentinel-1 GRDM data based on the complex amplitudes $\boldsymbol{z}$ of an ideal synthesized SAR image. The SAR system response $\mathbf{H}$ is modeled as a linear operator in the spatial domain, and it introduces spatial correlations due to possible zero-padding and spectral apodization during the SAR synthesis. A gain $\sqrt{\boldsymbol{a}} \in \mathbb{R}^{2N}_{+*}$ transforms amplitudes into digital numbers. Finally, a multi-looking and resampling step is performed in order to obtain approximately square pixels. This step is modeled

by a linear filtering operation by the operator $\mathbf{S}$. In equation, the generative model of figure 4.9 writes:

$$\widetilde{\boldsymbol{d}} = \mathbf{S}|\mathrm{diag}(\sqrt{\boldsymbol{a}})\mathbf{H}\boldsymbol{z}|^2\,, \tag{4.3}$$

where $\widetilde{\boldsymbol{d}}$ is a vector containing the GRDM intensities. In the following, GRDM images will be denoted with $\widetilde{\cdot}$ on the variables. $\mathbf{S}$ is an operator that multilooks then subsamples to transform single-look intensities into GRDM images. The squared modulus is applied separately to each complex value, and $\boldsymbol{z}$ is the vector formed by concatenation of the polarimetric complex amplitudes $\boldsymbol{z}_n$ at each pixel of the image.

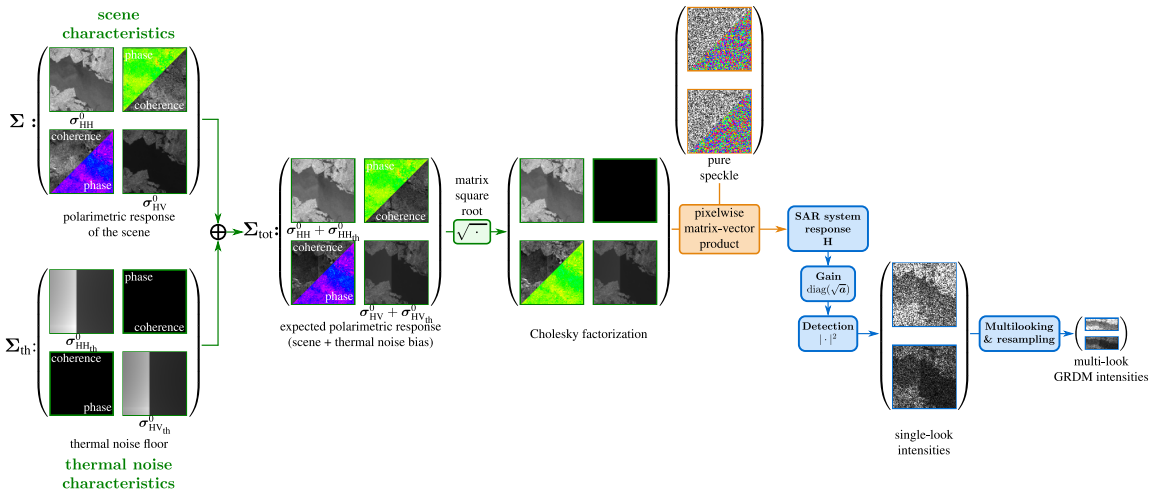


Figure 4.9: A generative model statistically equivalent to the physics of Sentinel-1 dual-pol GRD imagery: the effects of speckle and thermal noise are modeled through an equivalent covariance matrix $\mathbf{\Sigma}_{\mathrm{tot}}$.

The left-hand side of the figure illustrates the components of the polarimetric covariance matrix that characterizes the scene $\mathbf{\Sigma}$, the thermal noise floor $\mathbf{\Sigma}_{\mathrm{th}}$, and the total covariance matrix $\mathbf{\Sigma}_{\mathrm{tot}} = \mathbf{\Sigma} + \mathbf{\Sigma}_{\mathrm{th}}$.

Since thermal noise on the HH and HV measurements is independent, off-diagonal values of $\mathbf{\Sigma}_{\mathrm{th}}$ are null. While pure speckle follows a complex circular Gaussian distribution $\mathcal{N}_c(\mathbf{I})$ with a unitary covariance matrix $\mathbf{I}$, single-look complex polarimetric SAR images follow the distribution $\mathcal{N}_c(\mathbf{\Sigma}_{\mathrm{tot}})$.

We want to model the simulation of the speckle. Let us define the matrix $\mathbf{M}_{\mathrm{tot}}$ such that

$$\mathbf{M}_{\mathrm{tot}}\mathbf{M}_{\mathrm{tot}}^{*} = \mathbf{\Sigma}_{\mathrm{tot}}\,, \tag{4.4}$$

then pure speckle $\boldsymbol{\eta} \sim \mathcal{N}_c(\mathbf{I})$ can be turned into complex amplitudes by

$$\boldsymbol{z} = \mathbf{M}_{\mathrm{tot}}\boldsymbol{\eta} \quad \text{with } \boldsymbol{z} \sim \mathcal{N}_c(\mathbf{\Sigma}_{\mathrm{tot}}) \tag{4.5}$$

The factorization of matrix $\mathbf{\Sigma}_{\mathrm{tot}}$ in equation 4.4 provides a generative model of complex-valued polarimetric SAR amplitudes given in equation 4.5. In figure 4.9 we illustrate the Cholesky factorization even though other factorizations are also possible such as the computation of the square root of $\mathbf{M}_{\mathrm{tot}}$. We have the following factorization defined at a given pixel n:

$$\begin{aligned}\mathbf{\Sigma}_{\mathrm{tot}_n} &= \begin{pmatrix} \sigma^0_{\mathrm{HH}_n} + \sigma^0_{\mathrm{HH}_{\mathrm{th}\,n}} & \sqrt{\sigma^0_{\mathrm{HH}_n}\sigma^0_{\mathrm{HV}_n}}\rho_n e^{j\beta_n} \\ \sqrt{\sigma^0_{\mathrm{HH}_n}\sigma^0_{\mathrm{HV}_n}}\rho_n e^{-j\beta_n} & \sigma^0_{\mathrm{HV}_n} + \sigma^0_{\mathrm{HV}_{\mathrm{th}\,n}} \end{pmatrix} \\ &= \mathbf{M}_{\mathrm{tot}_n}\mathbf{M}^*_{\mathrm{tot}_n}\,, \end{aligned} \tag{4.6}$$

where $\rho_n \in [0,1[$ and $\beta_n \in [-\pi,\pi[$ are the polarimetric coherence and polarimetric phase at pixel n, and with

$$\mathbf{M}_{\mathrm{tot}_n} = \begin{pmatrix} \sqrt{\sigma^0_{\mathrm{HH}_n} + \sigma^0_{\mathrm{HH}_{\mathrm{th}\,n}}} & 0 \\ \sqrt{u_n}\exp(-j\beta_n) & \sqrt{\sigma^0_{\mathrm{HV}_n} + \sigma^0_{\mathrm{HV}_{\mathrm{th}\,n}} - u_n} \end{pmatrix} \tag{4.7}$$

where

$$u_n = \frac{\sigma^0_{\mathrm{HH}_n}\sigma^0_{\mathrm{HV}_n}}{\sigma^0_{\mathrm{HH}_n} + \sigma^0_{\mathrm{HH}_{\mathrm{th}\,n}}}\rho_n^2\,. \tag{4.8}$$

The full matrices $\mathbf{\Sigma}_{\mathrm{tot}}$ and $\mathbf{M}_{\mathrm{tot}}$ are block diagonal, with blocks $\mathbf{\Sigma}_{\mathrm{tot}_n}$ and $\mathbf{\Sigma}_{\mathrm{tot}_n}$, meaning that the polarimetric channels are correlated but pixels are independent.
The expression of this factorization will be useful to simulate synthetic speckle for the phase A of the training of the SAR2SAR network.
The polarimetric complex amplitudes $\boldsymbol{z}_n$ are formed by the pixelwise operations

$$\forall n \in [\![1, N]\!],\ \boldsymbol{z}_n = \mathbf{M}_{\mathrm{tot}_n}\boldsymbol{\eta}_n\,, \tag{4.9}$$

where n is a pixel index, $\mathbf{M}_{\mathrm{tot}_n}$ and $\boldsymbol{\eta}_n$ are defined at pixel n. $\mathbf{M}_{\mathrm{tot}_n}$ is formed by the matrix factorization

$$\begin{aligned}\mathbf{M}_{\mathrm{tot}_n}\mathbf{M}^*_{\mathrm{tot}_n} &= \mathbf{\Sigma}_{\mathrm{tot}_n} \\ &= \mathbf{\Sigma}_{n} + \mathbf{\Sigma}_{\mathrm{th}_n}\,. \end{aligned} \tag{4.10}$$

The bias due to the thermal noise floor can be removed using [69], leading to calibrated and corrected data $\widetilde{\boldsymbol{d}}_c$:

$$\widetilde{\boldsymbol{d}}_c = \mathrm{diag}(1/\widetilde{\boldsymbol{a}})\widetilde{\boldsymbol{d}} - \widetilde{\boldsymbol{\sigma}}^0_{\mathrm{th}}\,, \tag{4.11}$$

where the map of the inverse gain $1/\widetilde{\boldsymbol{a}}$ is obtained by pixelwise division and the gain $\widetilde{\boldsymbol{a}}$ and noise equivalent sigma zero $\widetilde{\boldsymbol{\sigma}}^0_{\mathrm{th}}$ at the GRDM resolution are obtained by application of the resampling operator $\mathbf{S}$:

$$\widetilde{\boldsymbol{a}} = \mathbf{S}\boldsymbol{a} \tag{4.12}$$

$$\widetilde{\boldsymbol{\sigma}}^0_{\mathrm{th}} = \mathbf{S}\boldsymbol{\sigma}^0_{\mathrm{th}} \tag{4.13}$$

$\widetilde{\boldsymbol{a}}$ and $\widetilde{\boldsymbol{\sigma}}^0_{\mathrm{th}}$ are made available by the space agencies in the metadata of SAR images.

The aim of the despeckling and thermal noise compensation is to remove both the thermal bias and the fluctuations due to thermal noise and speckle from the data $\widetilde{\boldsymbol{d}}$, meaning that we estimate

the expectation $\mathbb{E}[\widetilde{\boldsymbol{d}}_c]$. As shown below, this expectation corresponds to the reflectivity of the scene, up to the low-pass filtering effect of the SAR system. Combining equations (4.11) and (4.3) gives:

$$\mathbb{E}[\widetilde{\boldsymbol{d}}_c] = \operatorname{diag}\left(\frac{1}{\widetilde{\boldsymbol{a}}}\right) \mathbb{E}\left[\mathbf{S}|\operatorname{diag}(\sqrt{\boldsymbol{a}})\mathbf{H}\boldsymbol{z}|^2\right] - \widetilde{\boldsymbol{\sigma}}^0_{\text{th}} \tag{4.14}$$

Since calibration factors $\boldsymbol{a}$ vary slowly with the range and azimuth location, we can suppose

$$\mathbf{S}|\operatorname{diag}(\sqrt{\boldsymbol{a}})\mathbf{H}\boldsymbol{z}|^2 \approx \operatorname{diag}(\widetilde{\boldsymbol{a}})\mathbf{S}|\mathbf{H}\boldsymbol{z}|^2$$

leading to the simplification of the terms $\operatorname{diag}(1/\widetilde{\boldsymbol{a}})$ and $\operatorname{diag}(\widetilde{\boldsymbol{a}})$, and the expression of the expectation becomes

$$\mathbb{E}\left[|\mathbf{H}\boldsymbol{z}|^2\right] = \operatorname{diag}\left(\mathbf{H}\,\mathbb{E}\left[\boldsymbol{z}\boldsymbol{z}^*\right]\mathbf{H}^*\right)$$

where the notation diag() refers to the extraction of the diagonal of a square matrix. The linear operator $\mathbf{H}$ operates separately on each polarimetric channel. Moreover, complex amplitudes in $\boldsymbol{z}$ are uncorrelated for any pair of distinct pixels.
At pixel n, the expectation is thus equal to $\sum_k |\mathrm{H}_{nk}|^2 \cdot (\sigma^0_{\mathrm{HH}_k} + \sigma^0_{\mathrm{HH}_{\mathrm{th}\,k}})$ for HH polarimetric channel and $\sum_k |\mathrm{H}_{nk}|^2 \cdot (\sigma^0_{\mathrm{HV}_k} + \sigma^0_{\mathrm{HV}_{\mathrm{th}\,k}})$ for HV polarimetric channel, which corresponds to the low-pass filtered reflectivity obtained by accounting for the incoherent point spread function of the SAR system.
Provided that $\widetilde{\boldsymbol{\sigma}}^0_{\text{th}}$ matches the resampled and low-pass filtered noise equivalent sigma zero values in the HH and HV polarization channels such that

$$\widetilde{\boldsymbol{\sigma}}^0_{\text{th}} = \mathbf{S}|\mathbf{H}|^2\boldsymbol{\sigma}^0_{\text{th}}$$

then we obtain:

$$\mathbb{E}[\widetilde{\boldsymbol{d}}_c] \approx \mathbf{S}|\mathbf{H}|^2\boldsymbol{\sigma}^0\,, \tag{4.15}$$

where the square modulus $|.|^2$ is applied element-wise, leading to a linear operator $|\mathbf{H}|^2$ that can be interpreted as a convolution by the squared modulus of the complex-valued impulse response, for a shift-invariant SAR imaging system.

To conclude, our objective is to recover the scene $\widetilde{\boldsymbol{\sigma}}^0$, in ground range geometry, from the ground detected data $\widetilde{\boldsymbol{d}}$ by removing thermal bias and fluctuations due both to thermal noise and speckle. This scene is a low-pass filtered and resampled version of the scene in slant geometry $\boldsymbol{\sigma}^0$, meaning that we have:

$$\widetilde{\boldsymbol{\sigma}}^0 = \mathbf{S}|\mathbf{H}|^2\boldsymbol{\sigma}^0 \tag{4.16}$$

with $\mathbf{S}|\mathbf{H}|^2$ a linear operator that applies the incoherent point spread function of the system, low-pass filters and then resamples.

We want the network f_θ to predict an estimation of the despeckled corrected reflectivity. As shown in Figure 4.10, the distribution of the corrected images is different from the distribution of intensity images. The loss function is identical to the one used in 4.2.3 and introduced in equation 4.2, and the change compensation step is also performed in the same way.

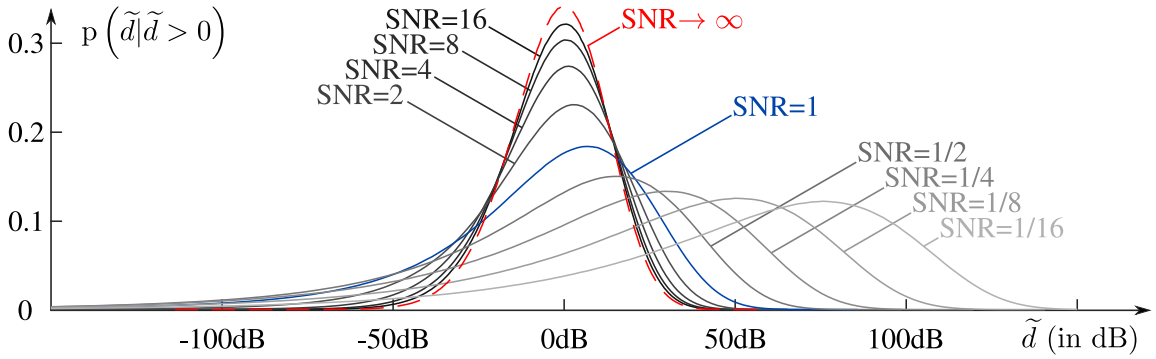


Figure 4.10: After compensation of the shift due to the thermal noise floor, corrected intensities follow a different statistical distribution. Here, the probability density function of the corrected intensities is represented, in log scale (dB), for different SNR values (ratios of the reflectivity and the thermal noise floor $\boldsymbol{\sigma}^0_{\text{th}}$). Compared to the Fisher-Tippett distribution followed in the absence of thermal noise correction (red dashed curve SNR $\to \infty$), the shape of the distribution is strongly modified by the correction, preventing from directly applying despeckling methods to corrected images.

The most straightforward way to reduce speckle and compensate for the bias due to thermal noise is a sequential processing: first despeckling of the intensity image leading to over-estimated reflectivities, then subtracting the thermal noise floor $\boldsymbol{\sigma}^0_{\text{th}}$. This strategy is illustrated in the top row of Fig.4.11. The main weakness of this strategy is that discontinuities of $\boldsymbol{\sigma}^0_{\text{th}}$ are present in the image processed by the despeckling algorithm. Imperfect restoration of these edges lead to artifacts after the subtraction step. A better approach would consist in removing the thermal noise bias before performing the reduction of fluctuations due to speckle and thermal noise.

4.3.2 Experimental results on simulated speckle

The first step of the training is done on simulated speckle. The training data set is the one used for out dual-polarimetric despeckling network in section 4.2.3.

As the images corrected with the Korosov algorithm can contain negative values, feeding the network with only the corrected images could be problematic. The thresholding is often done to get rid of the negative values, but it triggers a loss of information. The task is then harder for the network because some pixel information is lost. To make it easier, we decided to feed the network with not only the corrected image, but also the intensity image after calibration, and the correction σ_0^{th}. The input of the network is thus formed by the following vector:

$$\begin{pmatrix} \text{diag}(1/\widetilde{\boldsymbol{a}})\,\widetilde{\boldsymbol{d}} \\ \widetilde{\boldsymbol{d}}_c \\ \widetilde{\boldsymbol{\sigma}}^0_{\text{th}} \end{pmatrix}$$

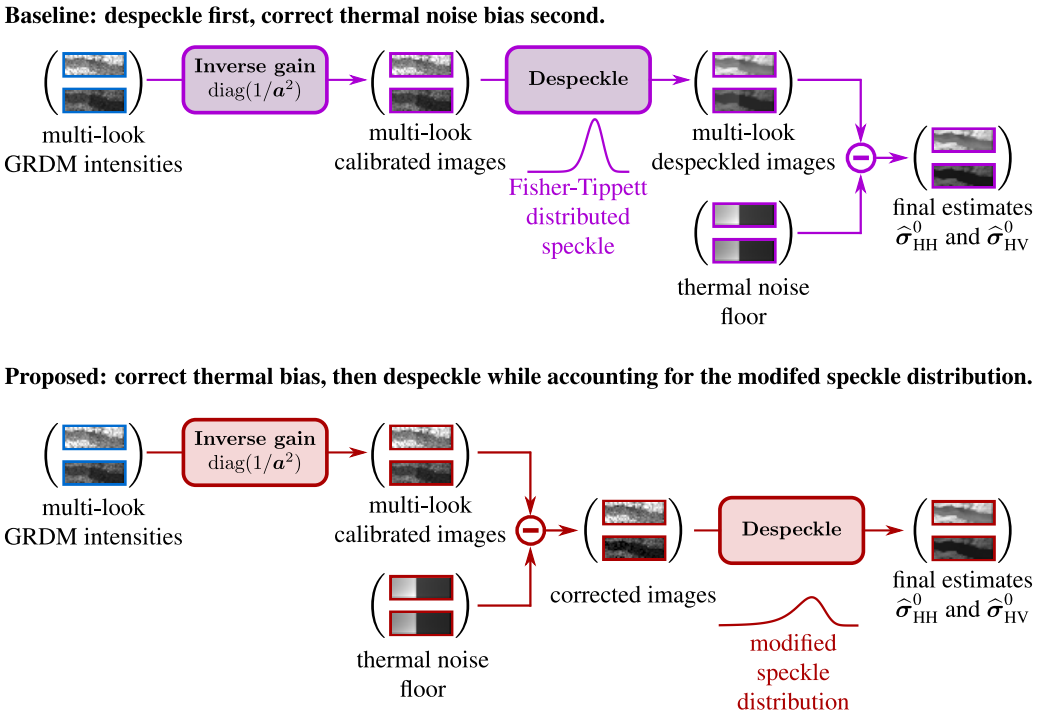


Figure 4.11: Despeckling techniques generally assume a multiplicative speckle model and must be applied *before* thermal noise bias removal (top). We propose to remove this bias before the despeckling step, provided that a specific despeckling technique be applied (bottom). This improves the restoration because the thermal noise floor discontinuities are removed before the despeckling step.

During the training phase A with simulated speckle, noisy images containing thermal noise are simulated using the statistically equivalent generative model for S1 EW mode GRD described in Figure 4.9. For simulating the thermal noise component, we randomly extract a patch from an actual thermal noise component from the training data set described in 4.2.2. The SNR factor, defined as the ratio between the ground truth reflectivity image and the thermal noise component, is randomly selected within the range $[0.1, 20]$ and leads to the adjustment of the reflectivity values to match the desired SNR. As the equivalent number of looks is not the same in all the image, the speckle is simulated based on the position of the extracted patch with the entire SAR image. The training hyperparameters are given in Table 4.1.

For testing, a new image is computed corresponding to an area in Belgica Bank. The image has been acquired by the Radarsat-2 satellite in C-band Fine quad polarimetric. To build a ground truth image, we first multilook the Single Look Complex (SLC) image to obtain roughly same ground pixel size matching S1 EW mode GRDM images. Then a despeckling was performed on the resulting covariance matrix using MuLog-BM3D [14]. The intensity information is extracted from the diagonal of the final despeckled covariance matrices.
The thermal noise component is then added and corresponds to the component of one of the Sentinel-1 image of our data set which contains a section of the first and the second sub-swath

from a reference image in the training data set described in section 4.2.2. The SNR factor for this simulation is fixed at 10. A set of 20 realizations of speckle are simulated and despeckled, and the Mean Squared Error is computed as shown in 4.2. Our approaches has lower MSE values (in linear and log-scale) than the dual-polarimetric despeckling network or the MuLog algorithm.

	HH polarization	**HV polarization**
MuLoG	0.000412 / *0.012417*	9.355e-05/ *0.03154*
Baseline	0.00038 / *0.01119*	9.122e-05 / *0.02614*
Proposed approach	0.00034 / *0.01037*	8.438e-05 / *0.01882*

Table 4.2: Mean Square Error computed between the restored image and the ground truth image in blue, and Mean Square Error computed between the log restored image and the log ground truth image in *green*. Both MSE are computed on 8 710 080 pixels (20 images of 688×633 pixels)

Our proposed approach leads to better results especially in areas where the values of the reflectivity are low. On the water, the fluctuations related to the thermal noise, are attenuated on the results in Figure 4.12 and Figure 4.13. There is no significant artifacts linked to the correction of the image after the despeckling (bright vertical lines are clearly visible on results obtained with the dual-polarimetric despeckling network at the boundary between sub-swaths).

4.3.3 Training on Sentinel-1 GRDM-EW images

In this section, we perform the phase B of the training of the SAR2SAR algorithm. The network is still trained using the loss in equation 4.2 evaluated on intensity images. The change compensation is also performed on the intensity images. The pre-estimated reflectivities used for this change-compensation step are estimated with the network trained in phase A in section 4.3.2.

Experimental results are given in Figure 4.14 on Sentinel-1 GRDM EW images from the river Ob in Russia. We compare ourselves to MuLoG, the SAR2SAR network and our dual-polarimetric joint despeckling network in section 4.2.3. For these three methods, the correction of the image is performed after the despeckling, leading to residual fluctuations where the reflectivity values are comparable to the thermal noise floor. As we are processing corrected images with our network, this is less visible in images restored with our approach.

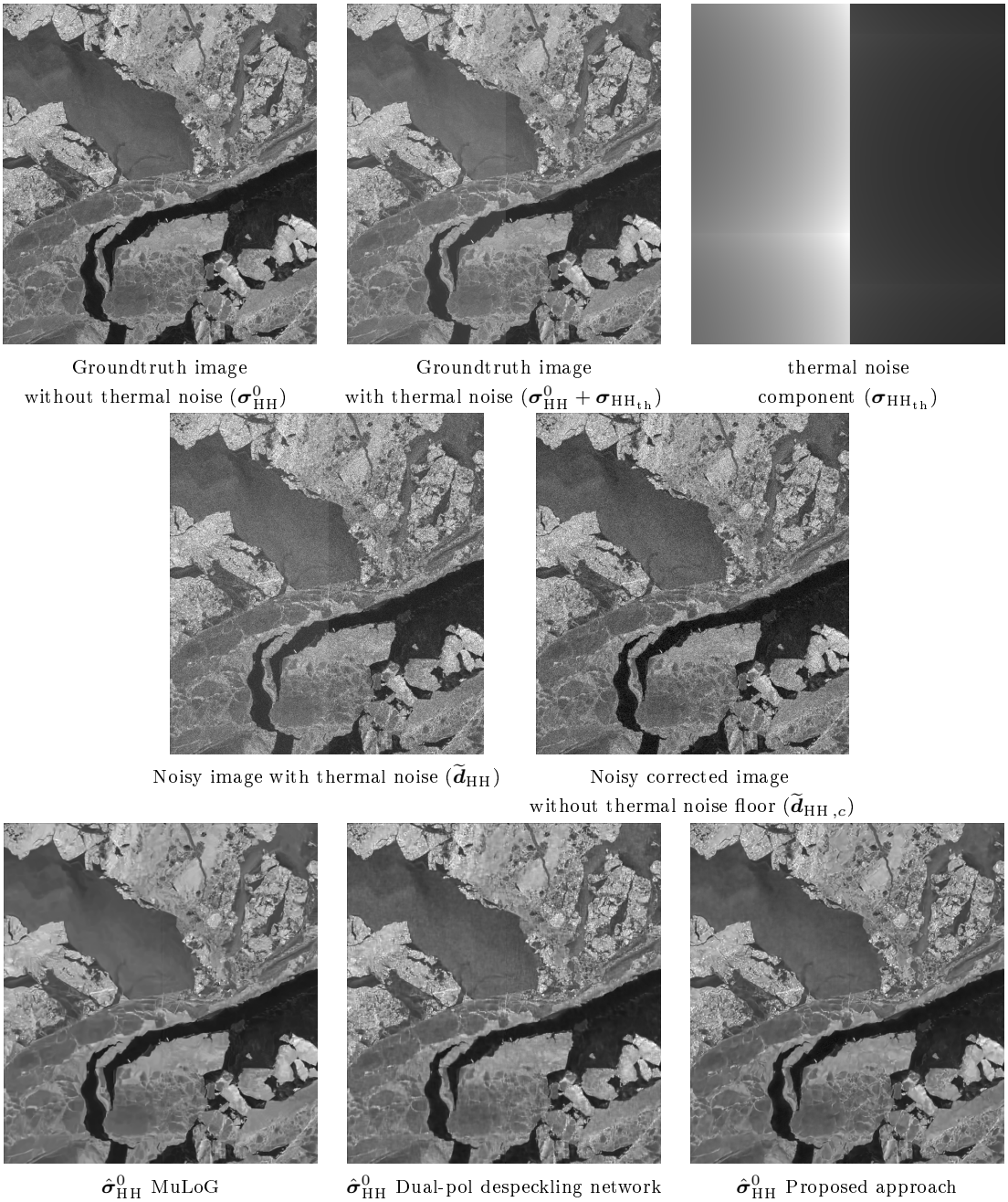
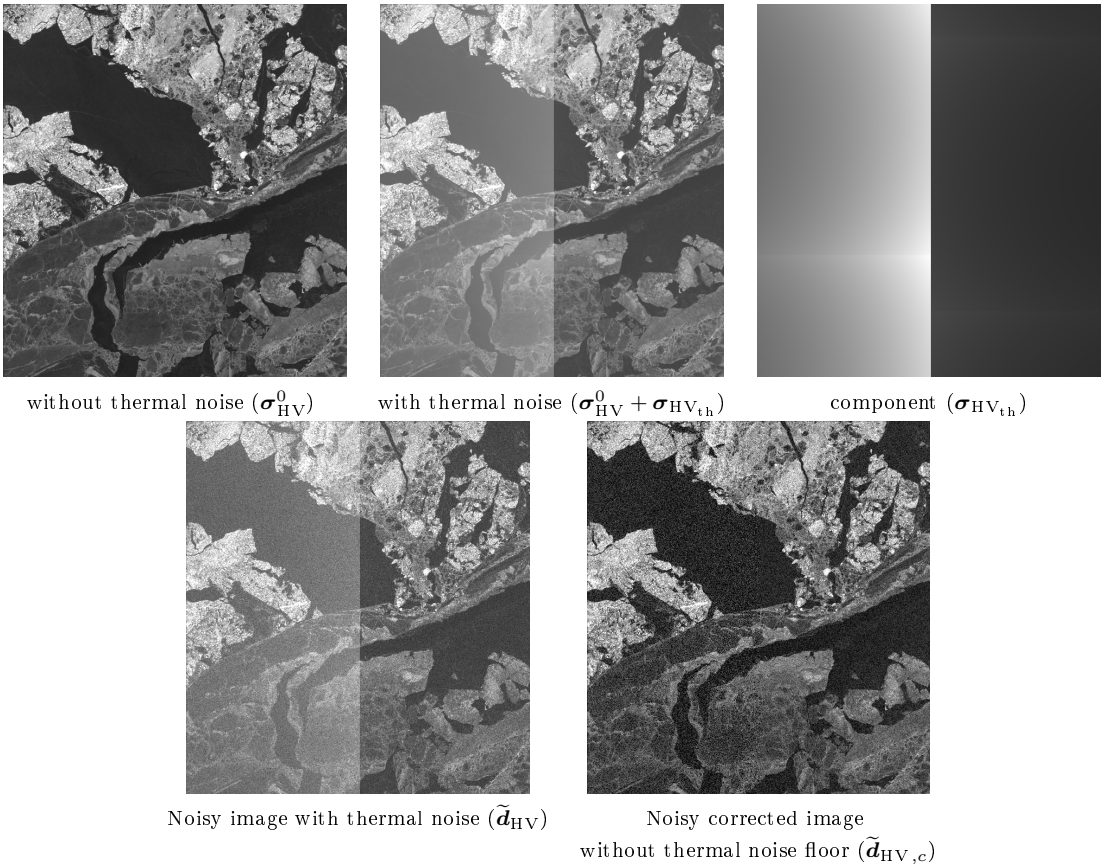


Figure 4.12: Despeckling results for the HH polarization. Speckle has been simulated and GRDM images computed based on the generative model in Figure 4.9. Fluctuations are attenuated in the water when the network is processing the corrected images. The vertical line coming from a correction posterior to despeckling, is less visible on the restored image by our proposed approach. Results on HV images are given in Figure 4.13.

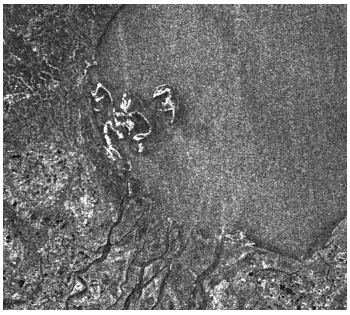


without thermal noise ($\boldsymbol{\sigma}^0_{\mathrm{HV}}$) with thermal noise ($\boldsymbol{\sigma}^0_{\mathrm{HV}} + \boldsymbol{\sigma}_{\mathrm{HV_{th}}}$) component ($\boldsymbol{\sigma}_{\mathrm{HV_{th}}}$)

Noisy image with thermal noise ($\widetilde{\boldsymbol{d}}_{\mathrm{HV}}$) Noisy corrected image without thermal noise floor ($\widetilde{\boldsymbol{d}}_{\mathrm{HV},c}$)

$\hat{\boldsymbol{\sigma}}^0_{\mathrm{HV}}$ MuLoG $\hat{\boldsymbol{\sigma}}^0_{\mathrm{HV}}$ Dual-pol despeckling network $\hat{\boldsymbol{\sigma}}^0_{\mathrm{HV}}$ Proposed approach

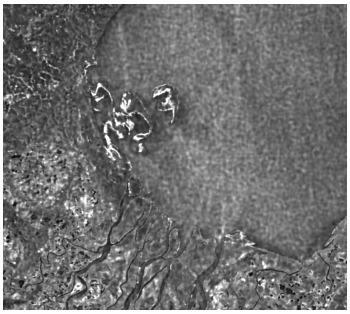
Figure 4.13: Despeckling results for HV polarization.



Noisy image (HH)

Despeckled with MuLog (HH)

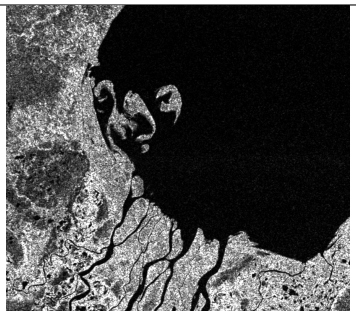
Despeckled with SAR2SAR (HH)



Despeckled with our dual-pol network (HH)

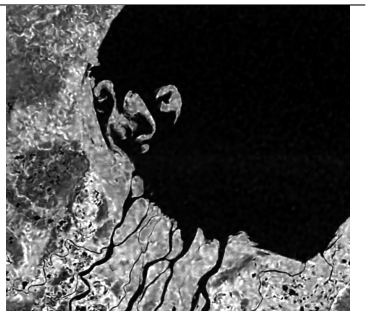
Despeckled with our approach phase A (HH)

Despeckled with our approach phase B (HH)

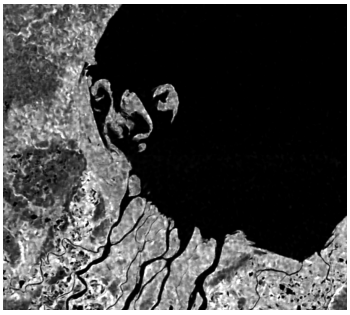


Noisy image (HV)

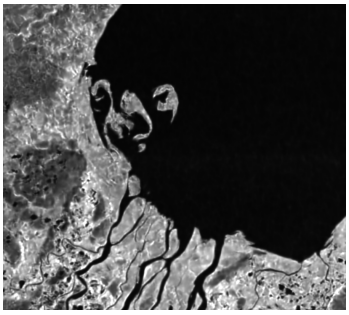
Despeckled with MuLog (HV)



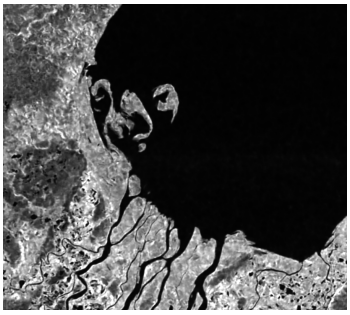
Despeckled with SAR2SAR (HV)



Despeckled with our dual-pol network (HV)



Despeckled with our approach phase A (HV)



Despeckled with our approach phase B (HV)

Figure 4.14: Result on Sentinel-1 GRDM-EW images, River Ob, Russia. The 300×300 pixels crop is located in the first sub-swath of the image. Our dual-polarimetric despeckling network introduced in section 4.2.3 is also used. The restored image with our method has more details than the MuLoG algorithm while reducing the fluctuations related to thermal noise that are still visible in the water in the dual-polarimetric and the SAR2SAR approaches.

4.4 Conclusion

In this work, we used the polarimetric information of Sentinel-1 GRDM EW images to improve despeckling. Because the main application of this work is the analysis of sea ice images, we have proposed a method combining the thermal noise removal and despeckling. The network trained in section 4.3.3 can be used on corrected SAR images which are widely used by the sea ice community. The experimental results show that we are able to reduce the fluctuations in low reflectivity areas that are caused by a level of thermal noise close to the values of reflectivity of ice and water.

Chapter 5

Multi-temporal despeckling

Publications related to this chapter:

- *Exploiting multi-temporal information for improved speckle reduction of Sentinel-1 SAR images by deep learning*, Emanuele Dalsasso, Ines Meraoumia, Loic Denis, Florence Tupin, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference), 2021**
  Recipient of the 2021 IEEE GRSS Symposium Prize Paper Award
- *Fast strategies for multi-temporal speckle reduction of Sentinel-1 GRD images*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Florence Tupin, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference), 2022**
- *Débruitage multi-temporel d'images radar à synthèse d'ouverture par apprentissage profond auto-supervisé*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Florence Tupin, **GRETSI conference, 2022**
- *Multi-temporal speckle reduction with self-supervised deep neural networks*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Remy Abergel, Florence Tupin, **IEEE Transactions on Geoscience and Remote Sensing, 2023**

In the previous chapter, the Multi-Input Multi-Output framework has been used for joint despeckling using the polarimetric information of a single GRD image. This approach has been elected because we had some constraints concerning the data: sea ice images have a lot of structural changes and motion through time. Providing the network with a temporal stack of images of the same area would not have been beneficial to the restoration process. However, in this chapter, the context of work is different: given that temporal stacks are acquired by satellites with only local changes (for the monitoring of urban areas for example), how can we use this temporal information to improve the despeckling performance?

The extraction and use of temporal information can be done **indirectly**, by temporal averaging and through the use of a surrogate of the multi-temporal stack (what we call *super-image* in this

manuscript, as in [16]); or **directly** by feeding the images to a network.

As it has been explained in Chapter 3, section 3.1, joint despeckling with a MIMO framework is not effective when the number of input images is higher than two. In the following work, we will be working in the Multi Input Single Output (MISO) framework.

This chapter will be structured as follows: in the first section, we will introduce multi-temporal despeckling and provide key references in the literature; in the second section, we will develop the proposed strategies based on temporal averaging and the computation of a super-image; and in the last section, we will present our multi-temporal deep learning based method directly exploiting the multi-temporal stack provided as input to the network.

5.1 Introduction to Multi-temporal despeckling

The Sentinel-1 satellite mission in the context of the Copernicus program of the European Space Agency aims at providing open source data of the whole planet Earth. The Sentinel-1A satellite was launched the third of April in 2014, and it started producing a huge volume of data with new images of a land area every 6 days. The images are freely provided online by the European Space Agency. In this context, we can work with multi-temporal stacks of SLC or GRD images provided by the Sentinel-1 satellite.

Thus, the community has been interested in developing multi-temporal despeckling to leverage the available volume of data. The spatial resolution of Sentinel-1 being between 5 meters to 40 meters approximately, the need for despeckling without changing the spatial resolution has been of great interest.

In 2001, long time before the Sentinel-1 launch, a multi-temporal despeckling method has been developed by Quegan [70]. The proposed filter consists in the following idea: starting from a multi-temporal averaging, we can improve the estimator by compensating for changes. The method relies on the length of the time series and the quality of the single-image restorations that are used for change compensation.

Successful single-image despeckling algorithms have also been extended to multi-temporal despeckling. The SAR-BM3D proposed by [39] is based on collaborative filtering of blocks of similar patches; in its multi-temporal extension MSAR-BM3D [54], the patches located at other dates in the stack are also considered when searching for similar patches.
The two-step multi-temporal non-local means [15], inspired by the iterative filtering proposed in [12], is based on a weighted averaging along the spatial but also the temporal dimensions of similar patches.

Adaptive filtering of SAR temporal stacks can also be done using an adaptive mean filter by computing the coefficient of variation and detect stable areas and areas containing changes through time [71], or by computing a change detection matrix based on responses of similarity cross tests between each pair of images selected within the stack [72, 73].
RABASAR [16] proposes to compute first a *super-image* by temporally multi-looking the image stack. This super-image has almost no residual speckle fluctuations. Then, the ratio image between the noisy image and the super-image is computed. The content of these ratio images is largely simplified and thus easier to restore because only speckle and changes with respect to the super-image are remaining. The final despeckled images are obtained after re-multiplication by the super-image. A drawback of ratio-based processing is that the lowest-contrasted structures present either in the speckled image or in the super-image might be improperly restored. This could lead to the suppression of details or the apparition of "ghost" structures leaking from the super-image as

explained in [74] where a quantitative comparison between the temporal arithmetic mean image and the temporal geometric mean image is done.

5.2 Simple integration of multi-temporal information with high-SNR average images for multi-temporal despeckling

Working with a multi-temporal stack of images, an easy way to extract and use the temporal information is to average all the images of the stack. Let $\boldsymbol{z}$ be the temporal stack of T SAR images noted $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_T$ and the corresponding intensity stack $\boldsymbol{i}$ such that $\boldsymbol{i}_1 = |\boldsymbol{z}_1|^2, \ldots, \boldsymbol{i}_T = |\boldsymbol{z}_T|^2$. The temporal mean in intensity $\overline{\boldsymbol{i}}$ is defined by

$$\overline{\boldsymbol{i}} = \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{i}_t \tag{5.1}$$

The level of noise of $\overline{\boldsymbol{i}}$ is very low.

We can even go further and work with a super-image. Its concept and computation has been introduced by [16], and directly flow from temporal averaging. The super-image is computed by despeckling the temporal mean $\overline{\boldsymbol{i}}$ with a despeckling algorithm. The resulting image, wich will be noted $\boldsymbol{i}_{\text{super}}$, does not represent the scene at any particular time as it averages the temporal information of the stack. These super-images are also used as a groundtruth for the supervised training steps of the despeckling SAR-CNN [17] and SAR2SAR [18] frameworks.

This subsection introduces simple approaches based on existing single-image despeckling algorithms to obtain multi-temporal despeckling methods. First, an extension of the Quegan filter is proposed to try to improve the results by using images despeckled with the SAR2SAR framework [18]; then an extension of RABASAR [16] is described using again the SAR2SAR network to despeckle the ratio images.

5.2.1 Quegan filter with reflectivities estimated with a deep learning based method

In this first proposed approach, we study a fast strategy to combine the best of single-image CNN despeckling while exploiting multi-temporal redundancy with the widely used Quegan filter. It especially focuses on Sentinel-1 GRD data, which are widely used for operational programs.

As a despeckling method, we consider the adaptation of SAR2SAR to GRD images. We recall that within SAR2SAR, a CNN is trained to estimate the reflectivity image $\boldsymbol{r}$ at each pixel using pairs of co-registered SAR images acquired at different dates. The framework has first been developed on single-look Sentinel-1 images. An adaptation to GRD data, presenting a different number of looks and spatially-varying speckle correlations, is presented in [75]. This network has been used as-is in the following.

Quegan filter [70] is a powerful yet simple approach to denoise multi-temporal stacks. When the temporal correlation between the images can be neglected it boils down to an average of change compensated images.

Let us consider a temporal stack of T intensity images where $\boldsymbol{i}_t$ and $\boldsymbol{r}_t$ denote the intensity and reflectivity of a specific date t respectively. The filtering formula giving the estimated reflectivity

$\hat{\boldsymbol{r}}_t^Q$ of date t is

$$\hat{\boldsymbol{r}}_t^Q = \frac{1}{T}\sum_{k=1}^{T} \hat{\boldsymbol{r}}_t^Q \ \frac{\boldsymbol{i}_k}{\hat{\boldsymbol{r}}_k} \tag{5.2}$$

The change compensation between date t and a date k of the multi-temporal stack is done thanks to an estimation of the reflectivity of each date k denoted by $\hat{\boldsymbol{r}}_k$. This is a tricky problem since in case of a perfect knowledge of $\hat{\boldsymbol{r}}_k$, the multi-temporal denoising would not be needed. Therefore, the better this estimation, the better the multi-temporal denoised result.

In practice, in the original paper, it is proposed to evaluate these estimates by local averages of the intensity values around the processed pixel for each date. This corresponds to a local spatial multi-looking. This estimation unsurprisingly leads to a loss of resolution, for instance blurring strong targets and edges, and has a negative impact on the global multi-temporal result.

A simple improvement is thus to replace these estimates by more efficient estimations, for instance provided by SAR2SAR trained on GRD images [75]. Experimental results are shown in Fig.5.1 on a stack constituted by 17 Sentinel-1 GRD images acquired around the town of Mallacoota, Australia. The figure shows more in details the improvement brought by integrating SAR2SAR within the multi-temporal Quegan filter. Replacing the spatial multilooking by the SAR2SAR network leads to a better preservation of details.

The main advantage of this method is that it is really easy to use. People in the remote-sensing community use GRD images because their geometry makes them easy to interpret and the level of speckle is lower than the one observable in Single Look images. Besides, Quegan filter is still widely used despite numerous available multi-temporal filters developed by the signal and image processing community. It can also be implemented with a low computational complexity by pre-computing the averages of multi-temporal data and storing them for the processing of many newly acquired dates.

5.2.2 Extension of Ratio-based despeckling (RABASAR)

The second strategy that has been considered is the ratio-based approach proposed in [16]. The idea is to create a super-image $\boldsymbol{i}_{\text{super}}$, as introduced at the beginning of this section 5.1, and to denoise a residual image corresponding to the ratio $\boldsymbol{\tau}_t$ between a specific date t and the associated super-image such that for each pixel k we have

$$\tau_{t,k} = \frac{i_{t,k}}{i_{\text{super},k}} \tag{5.3}$$

The ratio image is then despeckled using any despeckling algorithm. The statistics of the ratio image are not exactly the same as the noisy image and an adapted denoising algorithm is proposed in the original paper [16]. In this case, there is no change compensation before averaging as in Quegan strategy, but the residual image is easier to denoise than the original one as it has an improved stationarity.
Finally, the denoised estimation of $\boldsymbol{r_t}$ is retrieved by multiplying $\hat{\boldsymbol{\tau}}_t$ with the super-image $\boldsymbol{i}_{\text{super}}$:

$$\forall k \quad \hat{r}^R_{t,k} = \hat{\tau}_{t,k} \times i_{\text{super},k} \tag{5.4}$$

We propose to use SAR2SAR for the denoising of the ratio image $\boldsymbol{\tau}_t$ of date t. Note that this method can be applied to Single Look images but also GRD images, the only condition to be verified

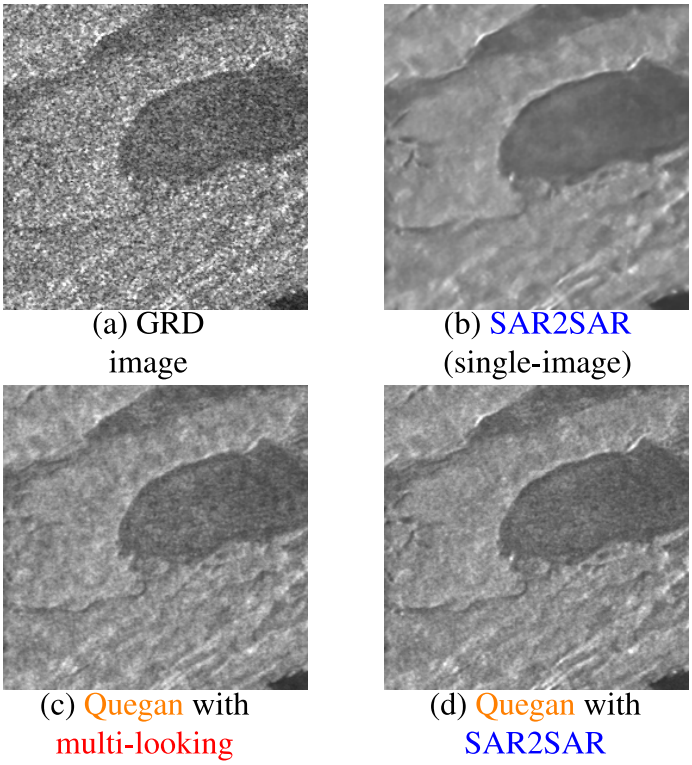


Figure 5.1: Details preservation and image quality is improved when SAR2SAR is integrated within Quegan filter (fig.(d)). On the one hand, single-image restoration results in a loss of fine structures (fig.(b)). On the other hand, the Quegan filter requires a high-quality speckle reduction algorithm: indeed, pre-estimating the reflectivites with a spatial averaging blurs some of the image details (fig.(c)).

is that the despeckling network has been trained on data from the same sensor and modality. Using SAR2SAR avoids sub-sampling altogether which preserves the spatial resolution of the restored images.

Owing to their non-linear nature, neural networks are very sensitive to the dynamic range of their inputs. Significantly shifting the dynamic range of input images between training and testing most often leads to catastrophic results. Ratio images have very different ranges compared to SAR intensity images: in the absence of significant changes between the image and the super-image, the expected value of the ratio is 1. It is then necessary to appropriately rescale them in order to use the SAR2SAR network trained on SAR images (i.e., not specifically trained on ratio images). We experimented with several normalization strategies and describe the one that worked best. Let us suppose that we work with a temporal stack of T images and want to despeckle the SAR image at date t.

In order to preserve the original range, in log domain, of the intensity image $\boldsymbol{i_t}$ when processing the ratio $\boldsymbol{\tau}$, we normalize the super-image $\boldsymbol{i}_{\text{super}}$ into $\boldsymbol{i}^0_{\text{super}}$ so its log-mean is equal to 0.

The modified ratio image $\boldsymbol{\tau}_0 = \boldsymbol{i}_t / \boldsymbol{i}^0_{\text{super}}$ is processed by SAR2SAR. The obtained despeckled ratio $\hat{\boldsymbol{\tau}}_0$ is then multiplied by the normalized super-image to produce the final estimate of the restored image $\hat{\boldsymbol{r}}$:

$$\hat{\boldsymbol{r}} = \hat{\boldsymbol{\tau}}_0 \times \boldsymbol{i}^0_{\text{super}}$$

We illustrate our method on single-look Sentinel-1 images of an area near Lelystad, Netherlands. A stack of 25 images was spatially co-registered and temporally averaged. Remaining speckle fluctuations were suppressed with MuLoG+BM3D [14], using an equivalent number of look estimated in a homogeneous area. A single-look amplitude image and the super-image are shown in Fig. 5.2, left column. Fig. 5.2 compares restoration results obtained by several strategies: the top block gives single-image restoration results and the bottom block shows how the use of a super-image improves the despeckling. Images (a) and (d) suffer from artifacts due to the application of a despeckling method that is sensitive to spatial correlations of speckle directly on a Sentinel-1 image. Downsampling the images reduces speckle correlation and suppresses these artifacts ((b) and (e)). This comes at the cost of a noticeable resolution loss, somewhat mitigated by the use of a super-image. SAR2SAR is robust to speckle correlations. It gives superior results in the single-image scenario (image c) and offers a restoration with an improved preservation of details such as thin roads or field edges using the proposed multi-temporal approach (image f).

In the previous approaches 5.2.1 and 5.2.2, multi-temporal information has been exploited in a simple way by working with temporal averaging and despeckling the ratio image between the noisy image and the super-image. The comparison between the fast strategy using the improved Quegan filter and the RABASAR adaptation with the SAR2SAR denoiser is given in Fig. 5.3.

Although easy to apply, these methods do not fully exploit the temporal information. In the following section, we propose a new deep learning based network taking a multi-temporal stack as input and thus extracting the temporal information on its own.

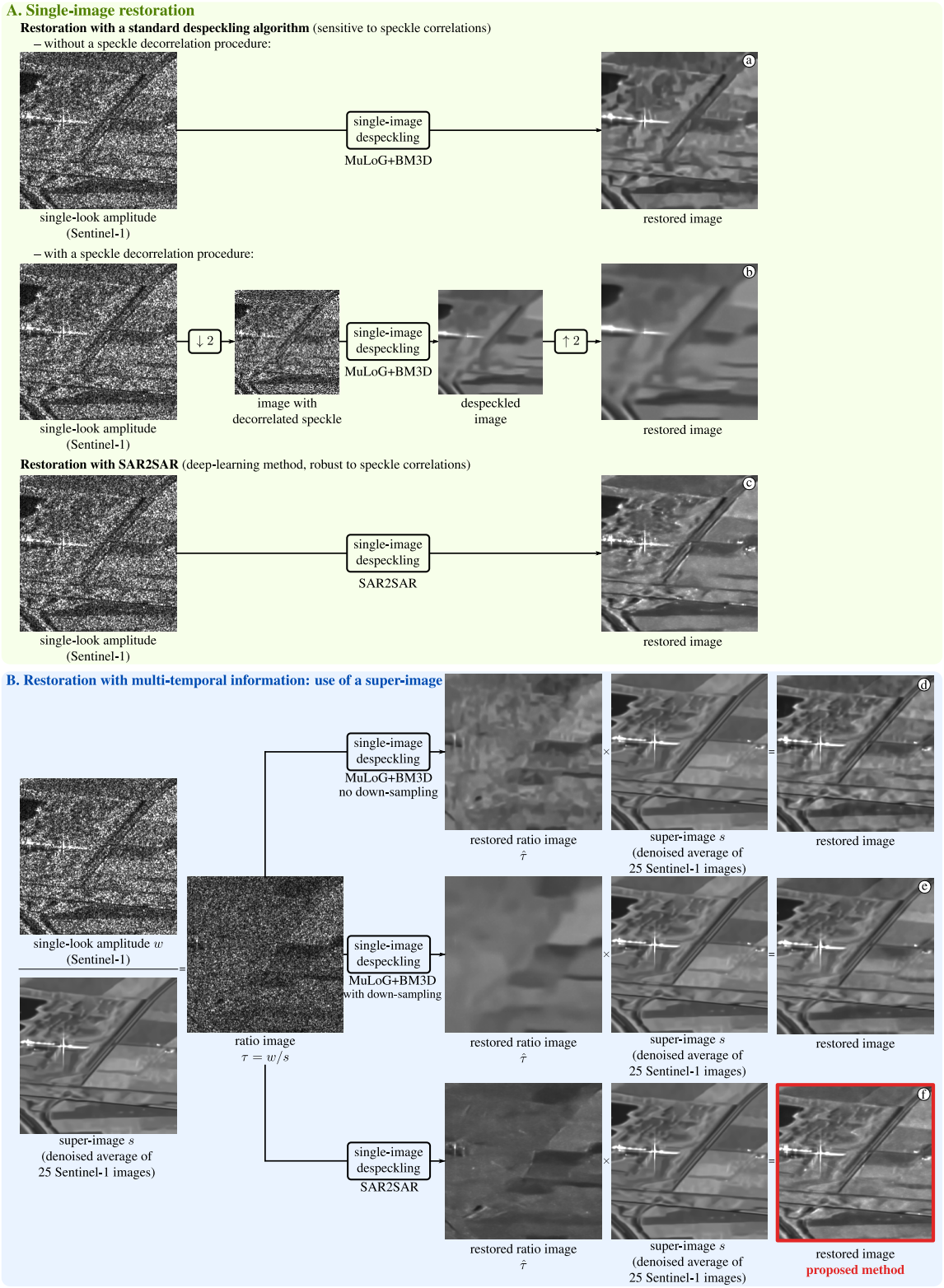


Figure 5.2: Comparison of several despeckling strategies for single-image and multi-temporal processing. Figure extracted from [67].

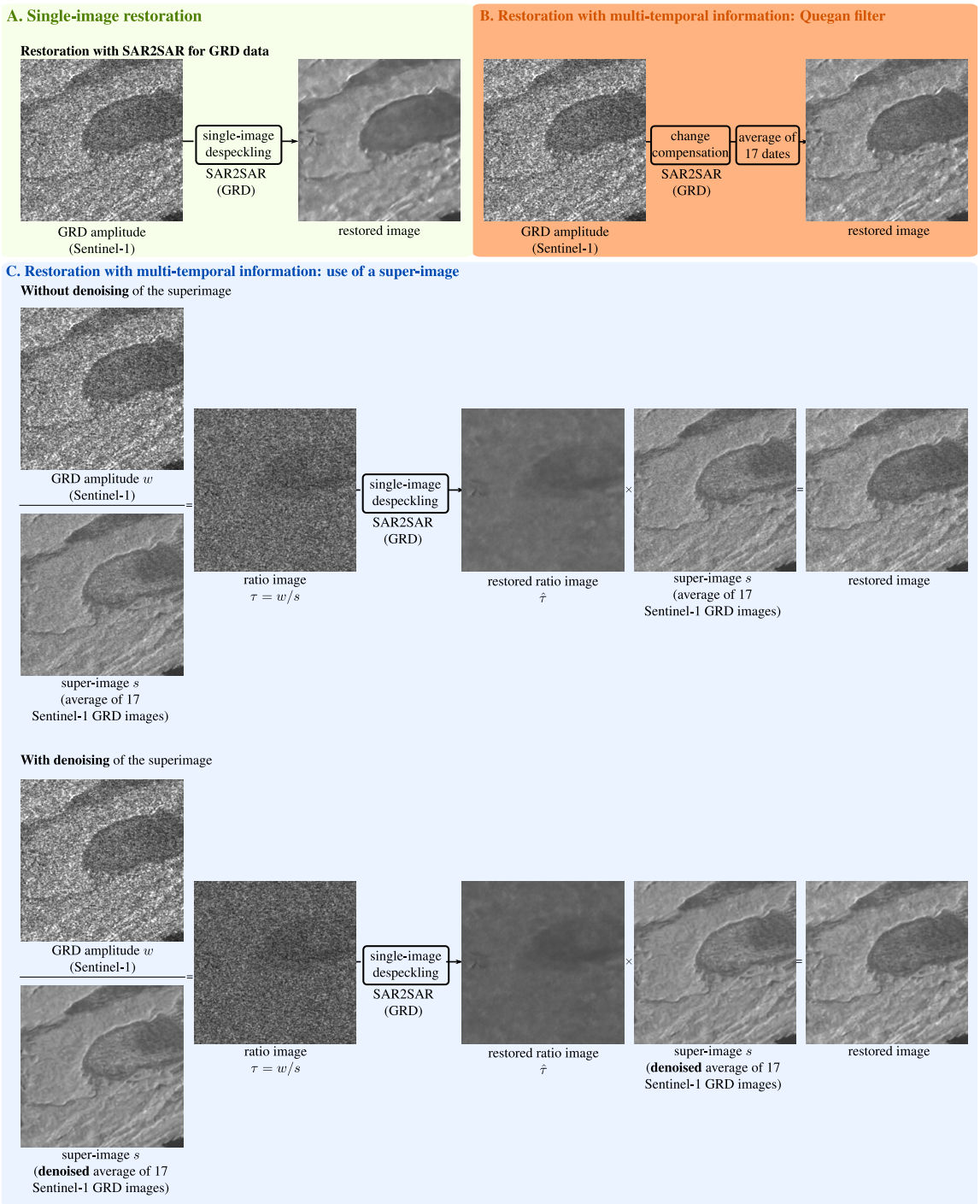


Figure 5.3: Comparison of several despeckling strategies for single-image and multi-temporal processing of GRD images. (A) Single image restoration: the SAR2SAR network is trained on GRD images and no temporal information is used. The estimated reflectivity are blurred and there is a lack of details. (B) Restoration with multi-temporal information and Quegan filter: despeckling based on temporal averaging with change compensated images. The SAR2SAR network is used to compute pre-estimated reflectivities to perform the change compensation. (C) Restoration with multi-temporal information and the use of super-image: the RABASAR based method. Two computations of the super-image are illustrated, the first one with simple temporal averaging and the second one with an additional denoising step of the temporal mean. The ratio image is computed and despeckled with SAR2SAR. With both computations of the super-image, the estimated reflectivities are sharper and contain more details. The higher the number of images in a stack, the fewer differences are between the two estimations. Figure extracted from [76].

5.3 Self-supervised joint multi-temporal despeckling technique

The network proposed in this section can be trained end-to-end to produce a despeckled image from a time series of co-registered SAR images. This is made possible by the use of a self-supervised loss function introduced recently by our research team [19], bypassing the impossibility to access high-quality ground truth images. Compared to simpler strategies based only on a single date enriched by a super-image (as introduced in section 5.2), we feed the network with all available dates. This leaves all freedom to the network to perform optimal temporal combinations, leading to improved restorations even when only a few additional images are included.

Our method is grounded on a generative model of speckle that accounts for **fully-developed speckle areas**, **the presence of dominant scatterers** due to man-made structures, **interferometric coherence**, and **the spatial correlations** induced by the SAR transfer function. This generative model will be developed in paragraph 5.3.1.

Reminders on the MERLIN method are given in section 5.3.2. The theoretical framework of the proposed method is developed in section 5.3. A numerical study is then performed on data with simulated speckle to characterize the performance of the method in section 5.3.5. The approach is then tested on stacks of TerraSAR-X Stripmap images in section 5.3.4. A study on the influence of the temporal correlations when despeckling has also been done in the last section.

5.3.1 A generative model for multi-temporal stacks of SAR images

The ability to partition the data into two mutually independent sets is central to our self-supervised training strategy. It is thus necessary to model the different sources of speckle correlations arising in multi-pass SAR imaging. If the images are acquired in interferometric conditions, then the speckle remains partially coherent from one pass to the next. Otherwise, the speckle is fully decorrelated and multi-temporal filtering can be very effective provided that changes in the scene remain limited (i.e. geometrical structures are aligned throughout the time series).

SAR images are a mix between:

(i) areas that follow Goodman's fully developed model (coherent summation of many similar elementary phasors). They are typically composed of rough surfaces and scattering volumes

(ii) regions where the complex amplitude is mainly defined by the magnitude and phase of dominant scatterers.

To include both phenomena, we build a composite model of a stack $\boldsymbol{z} \in \mathbb{C}^{TN}$ of T SLC SAR images, each with N pixels. The stack is described as the superimposition of two components: a *speckle component* $\boldsymbol{s} \in \mathbb{C}^{TN}$, driven by a reflectivity map $\boldsymbol{r} \in \mathbb{R}_{+*}^{TN}$, and the *dominant scatterers component* $\boldsymbol{d} \in \mathbb{C}^{TN}$, see Figure 5.4(a).

In the following, the multi-temporal stacks will be represented in the form of a column vector (e.g., $\boldsymbol{z} \in \mathbb{C}^{TN}$), by concatenation of the T images, and both the image at date t (noted $\boldsymbol{z}(t,\cdot) \in \mathbb{C}^N$, or $\boldsymbol{z}_t$ in compact form) and the vector of complex amplitudes at pixel k for all dates (noted $\boldsymbol{z}(\cdot,k) \in \mathbb{C}^T$) will be considered. A permutation matrix $\boldsymbol{\Pi}$ can be applied to transform the vector $\boldsymbol{z}$ from an ordering according to a scan of all pixels for each date, one date after another, to an

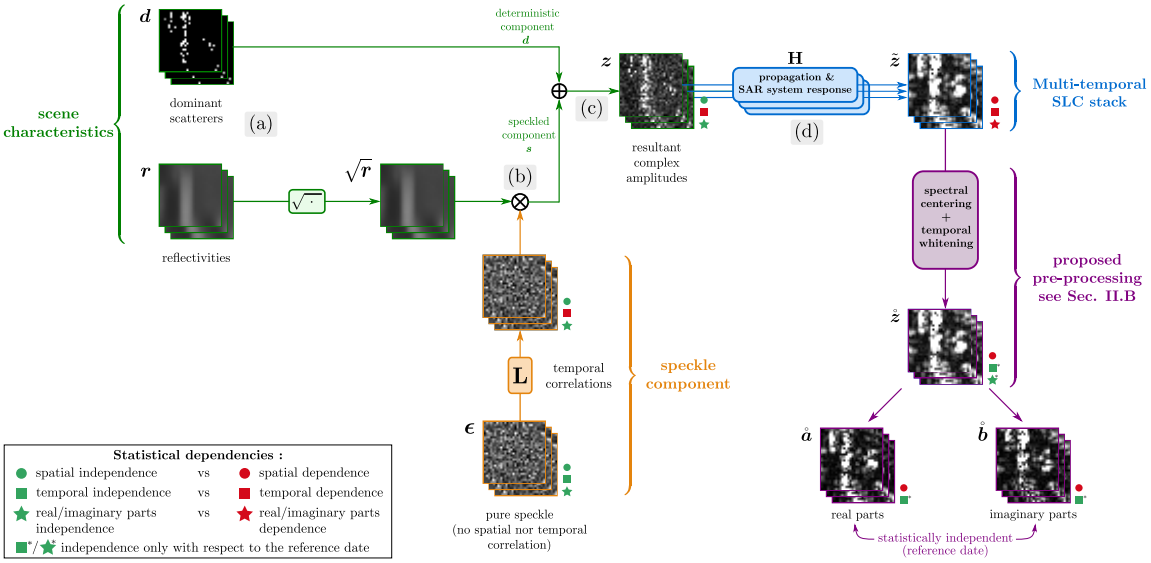


Figure 5.4: Generative model of speckle in multi-temporal SLC stacks of SAR images. Illustration extracted from [68].

ordering according to a scan of all dates for a given pixel, before moving to the next pixel:

$$\mathbf{\Pi} \boldsymbol{z} = \mathbf{\Pi} \begin{pmatrix} \boldsymbol{z}(t_1, \cdot) \\ \vdots \\ \boldsymbol{z}(t_T, \cdot) \end{pmatrix} = \begin{pmatrix} \boldsymbol{z}(\cdot, k_1) \\ \vdots \\ \boldsymbol{z}(\cdot, k_N) \end{pmatrix} . \tag{5.5}$$

This permutation will be handy to describe the structure of various covariance matrices.

According to Goodman's model [34], the *speckle component* $\boldsymbol{s}(\cdot, k) \in \mathbb{C}^T$ at pixel k follows a complex circular Gaussian distribution $\mathcal{N}_c(\mathbf{\Sigma}_k)$ defined by

$$\mathrm{p}(\boldsymbol{s}(\cdot, k) | \mathbf{\Sigma}_k) = \frac{1}{\pi^T \det(\mathbf{\Sigma}_k)} \exp\left[-\boldsymbol{s}(\cdot, k)^\dagger \, \mathbf{\Sigma}_k^{-1} \, \boldsymbol{s}(\cdot, k) \right], \tag{5.6}$$

where $\cdot^\dagger$ denotes the conjugate transpose; $\mathbf{\Sigma}_k$ is the speckle covariance matrix at pixel k. The matrix $\mathbf{\Sigma}_k$ can be written as

$$\mathbf{\Sigma}_k = \mathrm{diag}(\sqrt{\boldsymbol{r}(\cdot, k)}) \mathbf{\Gamma}_k \mathrm{diag}(\sqrt{\boldsymbol{r}(\cdot, k)})$$

with $\boldsymbol{r} \in \mathbb{R}_{+*}^{TN}$ the vector of reflectivities and $\mathbf{\Gamma}_k$ the coherence matrix. By definition, its entries verify $|\mathbf{\Gamma}_k(t_i, t_j)| \leq 1$ for all t_i and t_j and $\mathbf{\Gamma}_k(t, t) = 1$ for all t. Here, the square root function is applied entry-wise.

The coherence matrices characterize how the temporal evolution of the scene decorrelates the speckle. Starting from a pure speckle $\boldsymbol{\epsilon}_k \in \mathbb{C}^T$, with no correlation along the spatial and the temporal axis ($\boldsymbol{\epsilon}_k \sim \mathcal{N}_c(\mathbf{I})$), a multiplication by the matrix $\mathbf{L}_k$, where $\mathbf{L}_k \mathbf{L}_k^\dagger = \mathbf{\Gamma}_k$ (e.g., $\mathbf{L}_k$ is a Cholesky factor of coherence matrix $\mathbf{\Gamma}_k$), gives a random vector that follows the distribution $\mathcal{N}_c(\mathbf{\Gamma}_k)$.

Thus, the speckled component can be generated from $\boldsymbol{\epsilon}_k$ (Figure 5.4(b)):

$$\boldsymbol{s}(\cdot,k) = \mathrm{diag}(\sqrt{\boldsymbol{r}(\cdot,k)})\mathbf{L}_k\boldsymbol{\epsilon}_k\,. \tag{5.7}$$

The vector $\boldsymbol{s} \in \mathbb{C}^{TN}$ that concatenates all T images one after another can be obtained by

$$\boldsymbol{s} = \begin{pmatrix} \boldsymbol{s}(t_1,\cdot) \\ \vdots \\ \boldsymbol{s}(t_T,\cdot) \end{pmatrix} = \mathrm{diag}(\sqrt{\boldsymbol{r}})\underbrace{\boldsymbol{\Pi}^{-1}\begin{pmatrix} \mathbf{L}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{L}_N \end{pmatrix}\boldsymbol{\Pi}}_{\mathbf{L}}\boldsymbol{\epsilon}\,. \tag{5.8}$$

The covariance matrix of the *speckled component* $\boldsymbol{s}$ is block diagonal after a proper permutation

$$\mathrm{Cov}[\boldsymbol{s}] = \mathrm{diag}(\sqrt{\boldsymbol{r}})\boldsymbol{\Pi}^{-1}\begin{pmatrix} \boldsymbol{\Gamma}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \boldsymbol{\Gamma}_N \end{pmatrix}\boldsymbol{\Pi}\mathrm{diag}(\sqrt{\boldsymbol{r}})\,, \tag{5.9}$$

which shows that correlations are only along the temporal axis of the spatio-temporal stack.

The *dominant scatterers component* $\boldsymbol{d} \in \mathbb{C}^{TN}$ contains non zero values only at pixels with dominant scatterers. Such scatterers may appear or disappear at some point in the time series. The SLC amplitudes of the scene $\boldsymbol{z}$ then correspond to the superimposition of the two components: $\boldsymbol{z} = \boldsymbol{s} + \boldsymbol{d}$, see Figure 5.4(c). We model the effects of the atmospheric phase, the topographic (and possibly displacement) phase of the speckle component [77], and the spectral response of the SAR system as follows (Figure 5.4(d)):

$$\begin{aligned} \tilde{\boldsymbol{z}} &= \begin{pmatrix} \tilde{\boldsymbol{z}}(t_1,\cdot) \\ \vdots \\ \tilde{\boldsymbol{z}}(t_T,\cdot) \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{H}_1\mathrm{diag}(\exp(j\boldsymbol{\varphi}_1)) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{H}_T\mathrm{diag}(\exp(j\boldsymbol{\varphi}_T)) \end{pmatrix}\boldsymbol{z}\,, \end{aligned} \tag{5.10}$$

where $\tilde{\boldsymbol{z}}$ is the complex amplitude that includes these effects, $\mathbf{H}_t \in \mathbb{C}^{N\times N}$ is the SAR response for the t-th acquisition, and $\boldsymbol{\varphi}_t = \boldsymbol{\varphi}_{\mathrm{atmo}_t} + \boldsymbol{\varphi}_{\mathrm{topo}_t} + \boldsymbol{\varphi}_{\mathrm{disp}_t} \in \mathbb{C}^N$ combining the different sources of phase modification.

The spectral response of the SAR system is generally identical for all passes, up to a 2D shift due to angular discrepancies such as incidence and possibly squint angle differences between acquisitions. Linear operators $\mathbf{H}_t$, $1 \le t \le T$, can thus be written

$$\mathbf{H}_t = \mathrm{diag}(\exp(-j\boldsymbol{\psi}_t))\mathbf{Q}\mathrm{diag}(\exp(j\boldsymbol{\psi}_t))$$

where $\mathbf{Q} \in \mathbb{R}^{N\times N}$ is the real-valued operator applied in spatial domain and corresponding to a spectral response in Fourier domain. It is symmetrical and centered on the 0 frequency. The phase vector $\boldsymbol{\psi}_t$ is the 2D ramp corresponding to this 2D shift in Fourier domain accounting for the angular discrepancies at pass t. The complex amplitudes of the t-th pass can be rewritten

$$\tilde{\boldsymbol{z}}_t = \mathrm{diag}(\exp(-j\boldsymbol{\psi}_t))\mathbf{Q}\mathrm{diag}(\exp(j\boldsymbol{\varphi}_t + j\boldsymbol{\psi}_t))\boldsymbol{z}_t\,. \tag{5.11}$$

The linear operator $\mathbf{Q}$ accounts for the spectral apodization introduced to reduce the sidelobes of strong scatterers and a possible over-sampling which corresponds to a 0-padding in Fourier domain. Both of them induce a low-pass filtering effect on SAR images that does not depend on t.

Since the multi-temporal stack $\tilde{\boldsymbol{z}}$ is generated from $\boldsymbol{\epsilon}$ through a series of linear operations, $\tilde{\boldsymbol{z}}$ is also Gaussian-distributed with a mean equal to $\tilde{\boldsymbol{d}}$. For each date t and subvector $\boldsymbol{d}_t$, we can define the low-pass filtered dominant scatterers component $\tilde{\boldsymbol{d}}(t,\cdot) \in \mathbb{C}^N$ as

$$\tilde{\boldsymbol{d}}(t,\cdot) = \mathbf{H}_t \mathrm{diag}(\exp(j\boldsymbol{\varphi}_t))\boldsymbol{d}_t$$

The covariance matrix of the multi-temporal stack $\tilde{\boldsymbol{z}}$ can finally be derived

$$\begin{aligned}\mathrm{Cov}[\tilde{\boldsymbol{z}}] = &\begin{pmatrix} \mathrm{diag}(\exp(-j\boldsymbol{\psi}_1))\mathbf{Q}\mathrm{diag}(\exp(j\boldsymbol{\varphi}_1 + j\boldsymbol{\psi}_1)) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathrm{diag}(\exp(-j\boldsymbol{\psi}_T))\mathbf{Q}\mathrm{diag}(\exp(j\boldsymbol{\varphi}_T + j\boldsymbol{\psi}_T)) \end{pmatrix} \\ &\mathrm{diag}(\sqrt{\boldsymbol{r}})\boldsymbol{\Pi}^{-1}\begin{pmatrix} \boldsymbol{\Gamma}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \boldsymbol{\Gamma}_N \end{pmatrix}\boldsymbol{\Pi}\mathrm{diag}(\sqrt{\boldsymbol{r}}) \\ &\begin{pmatrix} \mathrm{diag}(\exp(-j\boldsymbol{\varphi}_1 - j\boldsymbol{\psi}_1))\mathbf{Q}^\dagger\mathrm{diag}(\exp(j\boldsymbol{\psi}_1)) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathrm{diag}(\exp(-j\boldsymbol{\varphi}_T - j\boldsymbol{\psi}_T))\mathbf{Q}^\dagger\mathrm{diag}(\exp(j\boldsymbol{\psi}_T)) \end{pmatrix}\end{aligned} \tag{5.12}$$

The complex values in the generated temporal stack $\tilde{\boldsymbol{z}}$ are both **spatially** and **temporally** correlated.

5.3.2 Reminder on the self-supervised single-image MERLIN method

The principle of the self-supervised training proposed in [19] called coMplex sElf-supeRvised despeckLINg (MERLIN), consists of splitting the real and imaginary components of a single-date SLC image and exploiting their statistical independence during the training phase. In the testing phase, two estimations are computed based on the real and imaginary parts. The final estimation is the average value of these two estimations. The training and testing strategies are described in Fig. 5.5.

We remind that two different tasks can be considered when extending speckle reduction to multi-temporal stacks:

(i) the Multiple Input Single Output (MISO) framework where several dates are provided in input but only a single image at a reference date t_{ref} is restored

(ii) the Multiple Input Multiple Output (MIMO) framework that restores at once all the dates provided in the input multi-temporal stack

As discussed in section 3.1, we follow the MISO approach depicted in Figure 5.6 for two reasons: the first one is the easier requirement of *statistical independence* with respect to the inputs of the network when a single output is considered. The second one is the computational constraint:

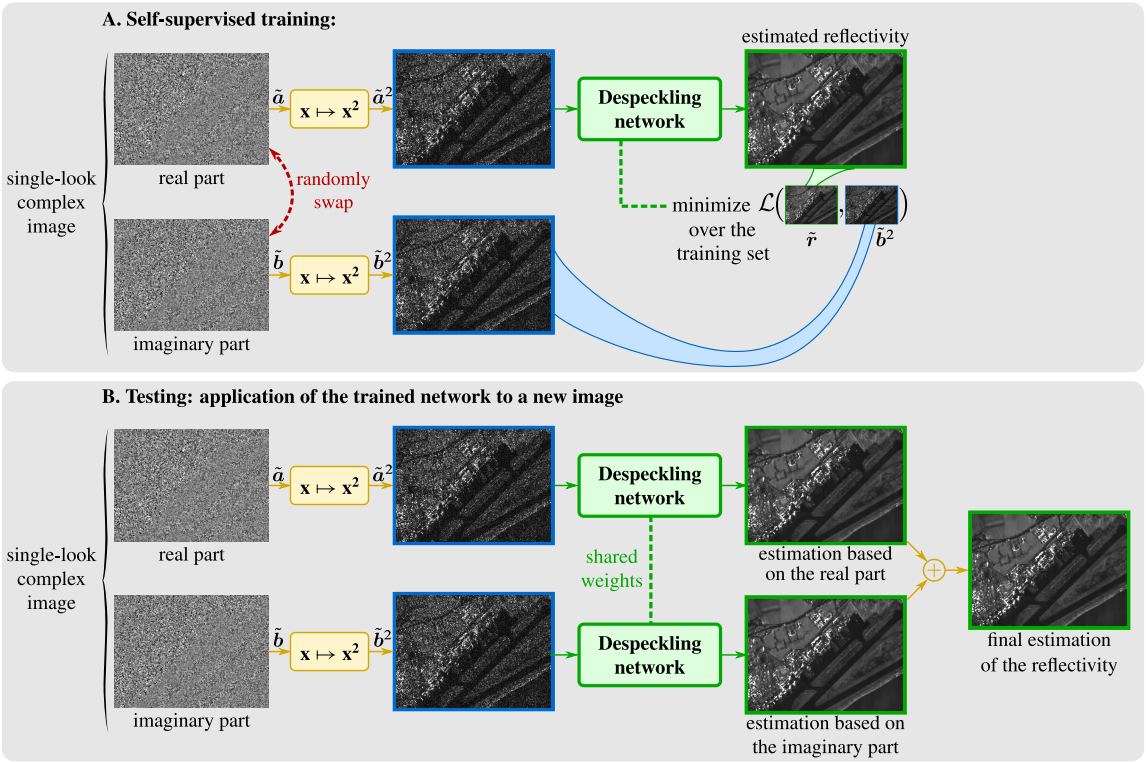


Figure 5.5: MERLIN strategy to despeckle one SLC image. (A) Self-supervised training: the complex values of the data are used to perform self-supervised despeckling. The real and imaginary parts are used as two independent noisy images of the same scene. A training based on Noise2Noise [21] uses the squared real part (or squared imaginary part) as input and supervises the training with the squared imaginary part (or the squared real part). (B) Testing and application of the trained network to a new image: to test the method on one SLC image, a first estimation of the reflectivity image is computed by feeding the squared real part to the network, and a second one by feeding it with the imaginary part. The final estimator is the average image between the two pre-estimations. Figure extracted from [19].

in order for a MIMO network to output very different images in case of large changes, several *independent paths* must emerge within the network architecture, which requires a *huge network capacity* [66] and a *careful initialization* to avoid getting stuck in poor quality local minima during training, as we observed in our preliminary experiments.

In our MISO multi-temporal approach, we provide the network with the multi-temporal SLC stack of T images where the real part (or imaginary part) of the reference date t_{ref} is excluded. This excluded component is then used to supervise the training under the assumption that it is statistically independent from the inputs. Here, the reflectivities $\boldsymbol{r}$ and dominant scatterers $\boldsymbol{d}$ are considered deterministic and only the speckle $\boldsymbol{\epsilon}$ is random. Two preprocessing steps are required to ensure this independence.

First, the shift of the SAR system response in the spectral domain at date t_{ref} induces correlations

between real and imaginary components at this date. Thus, the Hermitian symmetry of the SAR transfer function must be ensured. This issue was originally discussed in [19], it is extended to the more difficult case of a temporal stack in this work. A simple pre-processing step can be applied to recenter the spectrum of the image at the reference date around the 0 frequency by multiplication by the 2D phase ramp $\exp(j\boldsymbol{\psi}_{t_{\text{ref}}})$. In order to preserve interferometric coherence, we apply the same spectral shift to all dates such that the relative shift between Fourier spectra remains unchanged. We denote the centered complex amplitudes by $\dot{\boldsymbol{z}}$, defined by

$$\forall t,\ \dot{\boldsymbol{z}}(t,\cdot) = \text{diag}(\exp(j\boldsymbol{\psi}_{t_{\text{ref}}}))\tilde{\boldsymbol{z}}(t,\cdot) \tag{5.13}$$

where the phase ramp $\boldsymbol{\psi}_{t_{\text{ref}}}$ required to recenter the spectrum, can be estimated from the power spectrum of image $\tilde{\boldsymbol{z}}(t_{\text{ref}},\cdot)$. This leads to the following simplified expression at t_{ref}:

$$\dot{\boldsymbol{z}}(t_{\text{ref}},\cdot) = \mathbf{Q}\text{diag}(\exp(j\boldsymbol{\varphi}_{t_{\text{ref}}} + j\boldsymbol{\psi}_{t_{\text{ref}}}))\boldsymbol{z}_{t_{\text{ref}}}\,. \tag{5.14}$$

Second, a whitening step may be necessary to address the correlations along the temporal axis, depending both on the coherence matrices $\boldsymbol{\Gamma}_k$ (these matrices model how temporal decorrelations affect the scene) and the shifts induced by the phases $\boldsymbol{\psi}_t$ which model geometric decorrelation according to the interferometric baselines. In the context of multi-temporal speckle filtering, the stronger the correlations along the temporal dimension, the less useful the additional images. It is therefore recommended to consider time series with sufficient temporal speckle decorrelation for which no whitening step is necessary. If images are in interferometric configuration with a large coherence, a whitening step is required. Further experiments and a description of our proposed whitening procedure are presented in the section 5.3.5.

We will denote by $\mathring{\boldsymbol{z}}$ the stack after this preprocessing step, i.e., with minimal correlations along the temporal dimension. Please note that $\mathring{\boldsymbol{z}} = \dot{\boldsymbol{z}}$ in the absence of whitening step.
Assumption 1 summarizes that temporal correlations have been suppressed by the preprocessing step:

Assumption 1. *The preprocessed image $\mathring{\boldsymbol{z}}_{ref}$ at date t_{ref} is statistically independent of the images $\mathring{\boldsymbol{z}}_t$ for all dates $t \neq t_{ref}$.*

In our MISO framework, we will consider two sets of inputs (noted $\mathscr{E}_a$ and $\mathscr{E}_b$) that contain all images $\mathring{\boldsymbol{z}}_t$ except for the imaginary part $\mathring{\boldsymbol{b}}_{\text{ref}} \in \mathbb{R}^N$ (respectively the real part $\mathring{\boldsymbol{a}}_{\text{ref}} \in \mathbb{R}^N$) of $\mathring{\boldsymbol{z}}_{\text{ref}}$:

$$\mathscr{E}_a = \{\mathring{\boldsymbol{a}}_{\text{ref}}\} \cup \{\mathring{\boldsymbol{z}}_t | t \neq t_{\text{ref}}\} \text{ and } \mathscr{E}_b = \{\mathring{\boldsymbol{b}}_{\text{ref}}\} \cup \{\mathring{\boldsymbol{z}}_t | t \neq t_{\text{ref}}\}.$$

In the following proposition, we show that these inputs are independent from the component set aside. This independence will be key to train a network fed with the input set $\mathscr{E}_a$ (or $\mathscr{E}_b$) under the supervision of loss function involving the component $\mathring{\boldsymbol{b}}_{\text{ref}}$ and respectively $\mathring{\boldsymbol{a}}_{\text{ref}}$.

Proposition 1. *Under assumption 1, the input set $\mathscr{E}_a$ is statistically independent from the imaginary part $\mathring{\boldsymbol{b}}_{ref}$ at date t_{ref}, and similarly the input set $\mathscr{E}_b$ is statistically independent from the real part $\mathring{\boldsymbol{a}}_{ref}$.*

Proof. Under assumption 1, the image $\mathring{\boldsymbol{z}}_{\text{ref}}$ is independent from all other images $\mathring{\boldsymbol{z}}_t$ with $t \neq t_{\text{ref}}$. It remains to prove that the real and imaginary parts at time t_{ref} are independent. According to our

generative model of Sec.5.3.1, they can be expressed in terms of the speckle $\boldsymbol{\epsilon}_{\text{ref}}$ and the dominant scatterers $\boldsymbol{d}_{\text{ref}}$

$$\begin{pmatrix} \mathring{\boldsymbol{a}}_{\text{ref}} \\ \mathring{\boldsymbol{b}}_{\text{ref}} \end{pmatrix} = \begin{pmatrix} \Re(\dot{\boldsymbol{d}}_{\text{ref}}) \\ \Im(\dot{\boldsymbol{d}}_{\text{ref}}) \end{pmatrix} + \mathbf{M} \begin{pmatrix} \Re(\boldsymbol{\epsilon}_{\text{ref}}) \\ \Im(\boldsymbol{\epsilon}_{\text{ref}}) \end{pmatrix}, \tag{5.15}$$

where

$$\dot{\boldsymbol{d}}_{\text{ref}} = \mathbf{Q}\text{diag}(\exp(j\boldsymbol{\varphi}_{\text{ref}} + j\boldsymbol{\psi}_{\text{ref}}))\boldsymbol{d}_{\text{ref}} \tag{5.16}$$

and

$$\mathbf{M} = \begin{pmatrix} \mathbf{Q}\text{diag}\big(\cos(\boldsymbol{\alpha}_{\text{ref}})\sqrt{\boldsymbol{r}_{\text{ref}}}\big) & -\mathbf{Q}\text{diag}\big(\sin(\boldsymbol{\alpha}_{\text{ref}})\sqrt{\boldsymbol{r}_{\text{ref}}}\big) \\ \mathbf{Q}\text{diag}\big(\sin(\boldsymbol{\alpha}_{\text{ref}})\sqrt{\boldsymbol{r}_{\text{ref}}}\big) & \mathbf{Q}\text{diag}\big(\cos(\boldsymbol{\alpha}_{\text{ref}})\sqrt{\boldsymbol{r}_{\text{ref}}}\big) \end{pmatrix}$$

with $\boldsymbol{\alpha}_{\text{ref}} = \boldsymbol{\varphi}_{\text{ref}} + \boldsymbol{\psi}_{\text{ref}}$ and where the square root as well as the multiplications between vector $\sqrt{\boldsymbol{r}_{\text{ref}}}$ and the cosine and sine are all applied entry-wise.
Given that $\Re(\boldsymbol{\epsilon}_{\text{ref}})$ and $\Im(\boldsymbol{\epsilon}_{\text{ref}})$ are independent and identically distributed according to a Gaussian distribution $\mathcal{N}(\mathbf{0}, \frac{1}{2}\mathbf{I})$, the real-valued vector formed by the real and imaginary components is also distributed according to a Gaussian distribution:

$$\begin{pmatrix} \mathring{\boldsymbol{a}}_{\text{ref}} \\ \mathring{\boldsymbol{b}}_{\text{ref}} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \Re(\dot{\boldsymbol{d}}_{\text{ref}}) \\ \Im(\dot{\boldsymbol{d}}_{\text{ref}}) \end{pmatrix}, \tfrac{1}{2}\mathbf{M}\mathbf{M}^{\dagger} \right) \tag{5.17}$$

with

$$\mathbf{M}\mathbf{M}^{\dagger} = \begin{pmatrix} \mathbf{Q}\text{diag}(\boldsymbol{r}_{\text{ref}})\mathbf{Q}^{\dagger} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}\text{diag}(\boldsymbol{r}_{\text{ref}})\mathbf{Q}^{\dagger} \end{pmatrix}. \tag{5.18}$$

This shows that $\mathring{\boldsymbol{a}}_{\text{ref}}$ and $\mathring{\boldsymbol{b}}_{\text{ref}}$ are both jointly Gaussian and decorrelated, and thus, independent. □

5.3.3 Self-supervised training strategy

In the MERLIN framework [19], the following single-date loss function has been introduced:

$$\mathcal{L}_{\text{MERLIN}}(\boldsymbol{a}, \boldsymbol{u}) = \sum_k \frac{1}{2} \log u_k + \frac{a_k^2}{u_k}. \tag{5.19}$$

where the sum is computed over all the pixels k, $\boldsymbol{a}$ is the real part of a SLC image, and $\boldsymbol{u}$ is the output of the network based on the imaginary part $\boldsymbol{b}$.
This loss was applied to train a network fed with the imaginary part $\boldsymbol{b}$ of a single SLC image and supervised by the corresponding real part $\boldsymbol{a}$, or conversely by providing $\boldsymbol{a}$ to the network and supervising with $\boldsymbol{b}$. Assuming that $\boldsymbol{a}$ and $\boldsymbol{b}$ are statistically independent, the network was shown to learn how to estimate the reflectivities.

We extend this loss to our multi-temporal MISO framework by replacing $\boldsymbol{a}$ with $\mathring{\boldsymbol{a}}_{\text{ref}}$ and $\boldsymbol{b}$ with $\mathring{\boldsymbol{b}}_{\text{ref}}$. The parameters $\boldsymbol{\theta}$ of our regression model $f_{\boldsymbol{\theta}}$, i.e. the deep neural network, can be learned by minimizing the following multi-temporal extension of the MERLIN loss function:

$$\arg\min_{\boldsymbol{\theta}} \mathbb{E}_{\substack{\mathring{\boldsymbol{b}}_{\text{ref}}|\boldsymbol{r},\boldsymbol{d} \\ \mathscr{E}_a|\boldsymbol{r},\boldsymbol{d}}} \left[\mathcal{L}_{\text{MERLIN}} \left(\mathring{\boldsymbol{b}}_{\text{ref}}, f_{\boldsymbol{\theta}}(\mathscr{E}_a) \right) \right] + \mathbb{E}_{\substack{\mathring{\boldsymbol{a}}_{\text{ref}}|\boldsymbol{r},\boldsymbol{d} \\ \mathscr{E}_b|\boldsymbol{r},\boldsymbol{d}}} \left[\mathcal{L}_{\text{MERLIN}} \left(\mathring{\boldsymbol{a}}_{\text{ref}}, f_{\boldsymbol{\theta}}(\mathscr{E}_b) \right) \right] \tag{5.20}$$

According to Proposition 1, the inputs of the network $\mathscr{E}_a$ or $\mathscr{E}_b$ are independent from the images $\mathring{\boldsymbol{b}}_{\text{ref}}$ and $\mathring{\boldsymbol{a}}_{\text{ref}}$ used in the loss. It is thus impossible for the network to predict the stochastic component in these images. One could argue that the output $\boldsymbol{u} = \boldsymbol{a}$ would minimize equation (5.19) but due to the stochastic nature of speckle, it cannot be guessed from the input of the network.
In the following proposition, we consider the family of all possible models $f_{\boldsymbol{\theta}}$ that map the input images to a single output image. We then discuss in the proof of Prop.3 the special case of a sub-family of models corresponding to a given parameterization of the regression model $f_{\boldsymbol{\theta}}$, which is, in our case, a fixed neural network architecture.

Proposition 2. *The expectation of the multi-temporal MERLIN loss function (5.20) is minimal with respect to the predictions $f_{\boldsymbol{\theta}}(\mathscr{E}_a)$ and $f_{\boldsymbol{\theta}}(\mathscr{E}_b)$ if and only if $f_{\boldsymbol{\theta}}(\mathscr{E}_a) = \tilde{\boldsymbol{r}}_{ref} + 2\Im(\dot{\boldsymbol{d}}_{ref})^2$ and $f_{\boldsymbol{\theta}}(\mathscr{E}_b) = \tilde{\boldsymbol{r}}_{ref} + 2\Re(\dot{\boldsymbol{d}}_{ref})^2$, where $\tilde{\boldsymbol{r}}_{ref}$ is the diagonal of covariance matrix $\mathbf{Q}\text{diag}(\boldsymbol{r}_{ref})\mathbf{Q}^\dagger$ and $\dot{\boldsymbol{d}}_{ref} = \mathbf{Q}\text{diag}(\exp(j\boldsymbol{\varphi}_{ref} + j\boldsymbol{\psi}_{ref}))\boldsymbol{d}_{ref}$.*

Proof. We start by expressing the values of the two expectations that appear in equation (5.20). They involve terms of the form

$$\mathbb{E}\left[\sum_k \frac{\mathring{\boldsymbol{a}}_{\text{ref}}(k)^2}{\boldsymbol{u}(k)}\right] \text{ and } \mathbb{E}\left[\sum_k \frac{\mathring{\boldsymbol{b}}_{\text{ref}}(k)^2}{\boldsymbol{v}(k)}\right]$$

where

$$\boldsymbol{u} = f_{\boldsymbol{\theta}}(\mathscr{E}_b) \text{ and } \boldsymbol{v} = f_{\boldsymbol{\theta}}(\mathscr{E}_a)$$

They can be rewritten

$$\mathbb{E}\left[\mathring{\boldsymbol{a}}_{\text{ref}}^\dagger \text{diag}\left(\frac{1}{\boldsymbol{u}}\right)\mathring{\boldsymbol{a}}_{\text{ref}}\right] = \text{Tr}\left\{\text{diag}\left(\frac{1}{\boldsymbol{u}}\right)\mathbb{E}\left[\mathring{\boldsymbol{a}}_{\text{ref}}\mathring{\boldsymbol{a}}_{\text{ref}}^\dagger\right]\right\}$$
$$\mathbb{E}\left[\mathring{\boldsymbol{b}}_{\text{ref}}^\dagger \text{diag}\left(\frac{1}{\boldsymbol{u}}\right)\mathring{\boldsymbol{b}}_{\text{ref}}\right] = \text{Tr}\left\{\text{diag}\left(\frac{1}{\boldsymbol{u}}\right)\mathbb{E}\left[\mathring{\boldsymbol{b}}_{\text{ref}}\mathring{\boldsymbol{b}}_{\text{ref}}^\dagger\right]\right\}$$

where $1/\boldsymbol{u}$ denotes an entry-wise inversion. By marginalization of the Gaussian distribution defined in (5.17), we obtain

$$\mathbb{E}[\mathring{\boldsymbol{a}}_{\text{ref}}\mathring{\boldsymbol{a}}_{\text{ref}}^\dagger] = \Re(\dot{\boldsymbol{d}}_{\text{ref}})\Re(\dot{\boldsymbol{d}}_{\text{ref}})^\dagger + \tfrac{1}{2}\mathbf{Q}\text{diag}(\boldsymbol{r}_{\text{ref}})\mathbf{Q}^\dagger$$
$$\mathbb{E}[\mathring{\boldsymbol{b}}_{\text{ref}}\mathring{\boldsymbol{b}}_{\text{ref}}^\dagger] = \Im(\dot{\boldsymbol{d}}_{\text{ref}})\Im(\dot{\boldsymbol{d}}_{\text{ref}})^\dagger + \tfrac{1}{2}\mathbf{Q}\text{diag}(\boldsymbol{r}_{\text{ref}})\mathbf{Q}^\dagger$$

This leads to:

$$\mathbb{E}_{\mathring{\boldsymbol{b}}_{\text{ref}}|\boldsymbol{r},\boldsymbol{d}}\left[\mathcal{L}_{\text{MERLIN}}(\mathring{\boldsymbol{a}}_{\text{ref}}, \boldsymbol{u})\right] = \sum_k \frac{1}{2}\log \boldsymbol{u}(k) + \frac{\Re(\dot{\boldsymbol{d}}_{\text{ref}}(k))^2 + \frac{1}{2}\tilde{\boldsymbol{r}}_{\text{ref}}(k)}{\boldsymbol{u}(k)} \tag{5.21}$$

$$\mathbb{E}_{\mathring{\boldsymbol{a}}_{\text{ref}}|\boldsymbol{r},\boldsymbol{d}}\left[\mathcal{L}_{\text{MERLIN}}(\mathring{\boldsymbol{b}}_{\text{ref}}, \boldsymbol{v})\right] = \sum_k \frac{1}{2}\log \boldsymbol{v}(k)\frac{\Im(\dot{\boldsymbol{d}}_{\text{ref}}(k))^2 + \frac{1}{2}\tilde{\boldsymbol{r}}_{\text{ref}}(k)}{\boldsymbol{v}(k)}. \tag{5.22}$$

A necessary condition for the expectations to be minimal is:

$$\frac{\partial}{\partial \boldsymbol{u}(k)}\mathbb{E}_{\mathring{\boldsymbol{a}}_{\text{ref}}|\boldsymbol{r},\boldsymbol{d}}\left[\mathcal{L}_{\text{MERLIN}}(\mathring{\boldsymbol{a}}_{\text{ref}}, \boldsymbol{u})\right] = 0 \;\Rightarrow\; \boldsymbol{u}(k) = \tilde{\boldsymbol{r}}_{\text{ref}}(k) + 2\Re(\dot{\boldsymbol{d}}_{\text{ref}}(k))^2 \tag{5.23}$$

$$\frac{\partial}{\partial \boldsymbol{v}(k)} \mathbb{E}_{\mathring{\boldsymbol{b}}_{\mathrm{ref}}|\boldsymbol{r},\boldsymbol{d}} \left[\mathcal{L}_{\mathrm{MERLIN}}(\mathring{\boldsymbol{b}}_{\mathrm{ref}}, \boldsymbol{v}) \right] = 0 \Rightarrow \boldsymbol{v}(k) = \tilde{\boldsymbol{r}}_{\mathrm{ref}}(k) + 2\Im(\dot{\boldsymbol{d}}_{\mathrm{ref}}(k))^2. \tag{5.24}$$

The second-order derivatives for the values of $\boldsymbol{u}(k)$ and $\boldsymbol{v}(k)$ given by equations (5.23) and (5.24)

$$\left. \frac{\partial^2 \mathbb{E}_{\mathring{\boldsymbol{a}}_{\mathrm{ref}}|\boldsymbol{r},\boldsymbol{d}} \left[\mathcal{L}_{\mathrm{MERLIN}}(\mathring{\boldsymbol{a}}_{\mathrm{ref}}, \boldsymbol{u}) \right]}{\partial \boldsymbol{u}(k)^2} \right|_{\boldsymbol{u}(k)=\tilde{\boldsymbol{r}}_{\mathrm{ref}}(k)+2\Re(\dot{\boldsymbol{d}}_{\mathrm{ref}}(k))^2} = \frac{1}{2(\tilde{\boldsymbol{r}}_{\mathrm{ref}}(k) + 2\Re(\dot{\boldsymbol{d}}_{\mathrm{ref}}(k))^2)^2} \tag{5.25}$$

$$\left. \frac{\partial^2 \mathbb{E}_{\mathring{\boldsymbol{b}}_{\mathrm{ref}}|\boldsymbol{r},\boldsymbol{d}} \left[\mathcal{L}_{\mathrm{MERLIN}}(\mathring{\boldsymbol{b}}_{\mathrm{ref}}, \boldsymbol{v}) \right]}{\partial \boldsymbol{v}(k)^2} \right|_{\boldsymbol{v}(k)=\tilde{\boldsymbol{r}}_{\mathrm{ref}}(k)+2\Im(\dot{\boldsymbol{d}}_{\mathrm{ref}}(k))^2} = \frac{1}{2(\tilde{\boldsymbol{r}}_{\mathrm{ref}}(k) + 2\Im(\dot{\boldsymbol{d}}_{\mathrm{ref}}(k))^2)^2} \tag{5.26}$$

are both strictly positive, which shows that the values of $\boldsymbol{u}(k)$ and $\boldsymbol{v}(k)$ correspond to a minimum. Since the solution to equations (5.23) and (5.24) is unique, we have identified the only minimum of the objective function. □

Proposition 3. *Minimization of the expectation of the multi-temporal MERLIN loss function leads to an unbiased estimator* $[f_{\boldsymbol{\theta}}(\mathscr{E}_a) + f_{\boldsymbol{\theta}}(\mathscr{E}_b)]/2$ *of the sum of the low-pass filtered reflectivities* $\tilde{\boldsymbol{r}}_{ref}$ *and of the intensity of the low-pass filtered dominant scatterers* $|\dot{\boldsymbol{d}}_{ref}|^2$ *at date* t_{ref}*, provided that* $f_{\boldsymbol{\theta}}$ *is sufficiently expressive (e.g., a deep neural network with sufficient width).*

Proof. Under the Universal Approximation Theorem for width-bounded ReLU networks [78], a network with sufficient width can be built to approximate an arbitrary and Lebesgue-integrable function $f_{\boldsymbol{\theta}}$. If less expressive estimators $f_{\boldsymbol{\theta}}$ are considered such as smaller networks, not fully-connected architectures or even other estimators than deep neural networks, a bias may appear. This is mainly due to the reduced ability of the estimator to match the optimal output given in Proposition 2.
For a sufficiently expressive estimator producing the optimal output, according to Proposition 2, the minimum of the expectation of the multi-temporal MERLIN loss function is reached for $f_{\boldsymbol{\theta}}(\mathscr{E}_a) = \tilde{\boldsymbol{r}}_{\mathrm{ref}} + 2\Im(\dot{\boldsymbol{d}}_{\mathrm{ref}})^2$ and $f_{\boldsymbol{\theta}}(\mathscr{E}_b) = \tilde{\boldsymbol{r}}_{\mathrm{ref}} + 2\Re(\dot{\boldsymbol{d}}_{\mathrm{ref}})^2$. The computation of the average concludes the proof:

$$\forall k,\ \frac{f_{\boldsymbol{\theta}}(\mathscr{E}_a)(k) + f_{\boldsymbol{\theta}}(\mathscr{E}_b)(k)}{2} = \tilde{\boldsymbol{r}}_{\mathrm{ref}}(k) + |\dot{\boldsymbol{d}}_{\mathrm{ref}}(k)|^2. \tag{5.27}$$

□

Figure 5.6 illustrates the principle of the self-supervised training introduced in Propositions 2 and 3: during training, we minimize MERLIN loss with the sets $\mathscr{E}_a$ or $\mathscr{E}_b$ as input and the images $\mathring{\boldsymbol{b}}_{\mathrm{ref}}$ or $\mathring{\boldsymbol{a}}_{\mathrm{ref}}$ in the supervision. This leads to optimal weights $\boldsymbol{\theta}^*$ at the end of the training phase. At test time, the estimates $f_{\boldsymbol{\theta}^*}(\mathscr{E}_a)$ and $f_{\boldsymbol{\theta}^*}(\mathscr{E}_b)$ are averaged to produce the final estimate.
For practical reasons, we use a convolutional U-Net architecture [79] which is also used in the original MERLIN method. We consider a limited number of images in the training phase and an approximate minimization based on stochastic gradient computed over mini-batches. The estimator $f_{\boldsymbol{\theta}^*}$ obtained is then only sub-optimal.

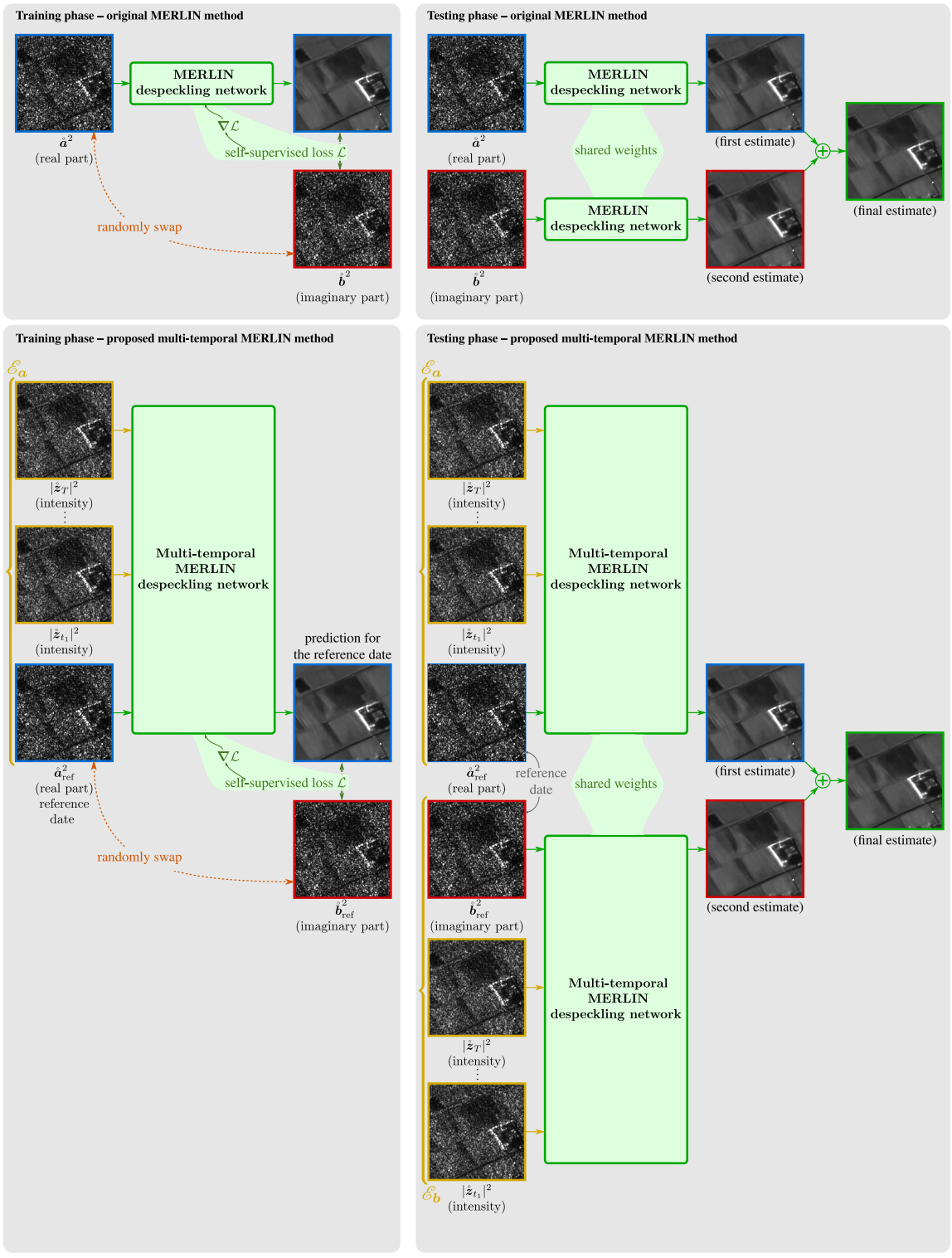


Figure 5.6: Principle of the self-supervised method MERLIN: original approach [19] (top row) and proposed multi-temporal extension (bottom row). For better visualization, the displayed images are amplitude images. Figure extracted from [68].

5.3.4 Experimental results

The performance of the proposed multi-temporal MERLIN strategy is first studied on images with simulated speckle. Results on Single Look Complex TerraSAR-X images are then presented. In both cases, we compare multi-temporal MERLIN networks trained for an increasing number of additional inputs to study the quality improvement brought by these additional dates.

Quantitative analysis on simulated speckle

The unsupervised learning strategy presented in Section 5.3.3 is motivated by the lack of speckle-free ground-truth images associated to each speckled SAR image. Yet, in order to perform a quantitative assessment of multi-temporal filtering, we first consider a simulated speckle framework in which both speckle-free and speckle-corrupted images are available. We build high-quality speckle-free stacks by multi-temporal filtering with RABASAR-SAR2SAR [67] presented in section 5.2.2. We then generate corrupted versions with simulated speckle corresponding to an ideal SAR transfer function, i.e. speckle with no spatial correlation in the simulated images. This reference data set is composed of 5 multi-temporal stacks of despeckled Sentinel-1 images, each stack containing from 25 to 69 images. Since the stacks are obtained from actual SAR images, realistic changes can be observed throughout the time series. To simplify the simulations, we assume fully-developed speckle: the ground-truth images correspond to the reflectivities $\boldsymbol{r}$ and no dominant scatterer is considered: $\boldsymbol{d} = \mathbf{0}$. Information on the training sets and the hyperparameters used in all our network trainings are gathered in table 5.1. The hyperparameters are kept unchanged whatever the number of additional inputs.

		Synthetic speckle Sentinel-1	**Actual speckle TerraSAR-X [Sentinel-1]**
# stacks		7	2 [1]
# images		237	52 [12]
avg images/stack		33.9	26 [12]
patch size		256×256	256×256
batch size		8	8
# patches		1616	576 [1568]
# batches		202	72 [196]
# epochs		1000	1000
learning rate	$\left\{\right.$	10^{-3}	10^{-3}
		10^{-4} after 10 epochs	10^{-4} after 10 epochs
	$\left.\right\}$	10^{-5} after 910 epochs	10^{-5} after 910 [860] epochs

Table 5.1: Training parameters of the multi-temporal MERLIN networks (number of input channels from 2 to 20)

We first evaluate the gain brought by the additional dates on the quality of the estimated speckle-free image. Depending on the presence or absence of change, including an additional input image may disturb or help the despeckling process. When comparing the performances of two networks, a network with fewer inputs that underwent less changes might be favored over a network with more inputs which were all impacted by larger changes. We mitigate the impact of this phenomenon on our analysis by evaluating the performance of our networks on combinations of additional dates forming nested sets. Thus, a network with j additional inputs with $j > i$ shares the same i additional dates as a smaller network with i additional inputs, but also benefits from $j-i$ supplementary inputs.

Figure 5.7 shows boxplots of the Peak Signal-to-Noise Ratio (PSNR) values computed on the log-reflectivities, for an increasing number of additional input images. The boxplots give for each configuration the minimum PSNR value; first, second, and third quartile PSNR values; and the maximum PSNR value. These statistics are computed over 88400 patches of 256×256 pixels, corresponding to different spatial locations, choices of dates included as input, or speckle realizations. The restoration quality, measured by the PSNR values, improves with the number of images. This improvement is largest when the first additional dates are included. Including a few more dates to an already large number of inputs produces a marginal improvement: unsurprisingly, multi-temporal filtering follows a law of diminishing returns with respect to the number of input dates.
Note that the dispersion of PSNR values for the mono-date filtering (leftmost boxplot of Figure 5.7) is very limited compared to the dispersion of PSNR values obtained with multi-temporal filtering. This is due to the variability of changes present in the additional channels. In multi-temporal filtering, situations with limited changes are more favorable to filtering and lead to better PSNR values, while drawing a set of dates with larger changes inevitably gives a worse PSNR value. The variable luck in how similar the additional dates were explains the PSNR fluctuations.
As illustrated by Figure 5.8, PSNR values improve when increasing the number of additional input images due to the joint reduction of the estimation bias and of the estimation variance. As illustrated by the bias term, additional channels help preserve the spatial resolution, reduce the blur around sharp structures such as points, lines and edges. By not only combining spatial samples but also temporal samples, the estimation variance is reduced by multi-temporal filtering.

The line profiles shown in Figure 5.9 confirm the improved ability to restore fine structures with multi-temporal filtering: processing a single date (green line) makes it difficult to retrieve the contrast of thin lines (edges at the border of fields); with an additional date, or even better, with 4 additional dates, these structures are much better restored.

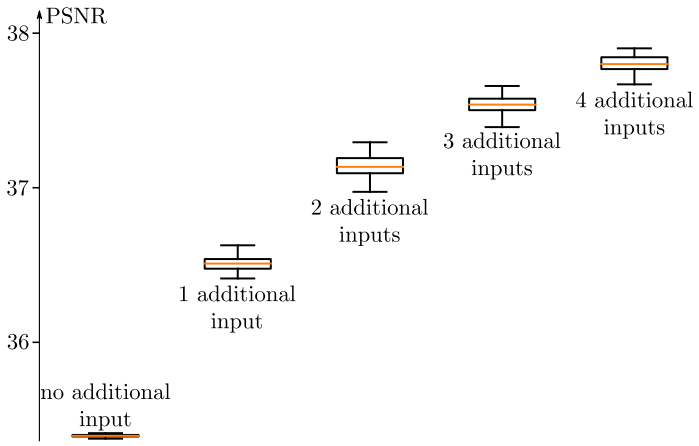


Figure 5.7: Boxplots of PSNR values obtained for different draws of additional dates and various speckle realizations (each box plot indicates the minimum value, first quartile, in orange: median value, third quartile, and maximum). Our multi-temporal MERLIN method outperforms the baseline methods in terms of PSNR: with 3 additional inputs, the median PSNR with MSAR-BM3D is 36.04 dB (-1.50 dB) and with 2SPPB it is 30.06 dB (-7.48dB).

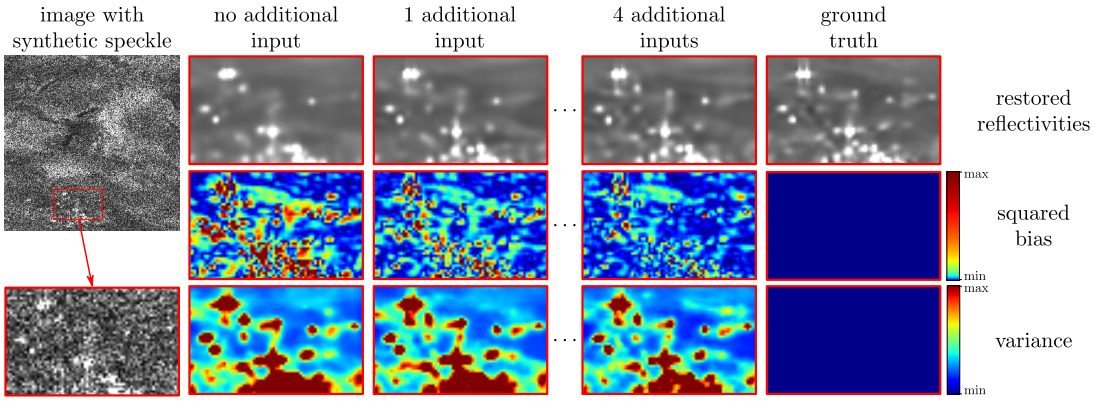


Figure 5.8: Squared bias and variance averaged over 100 multi-temporal MERLIN estimations of the reflectivities of a Sentinel-1 stack of Limagne (France). The speckle is simulated based on the method described in [67].

Results on actual Sentinel-1 Stripmap data and TerraSAR-X data

After the successful validation of our approach on time series with simulated speckle, we now turn to real speckle.
In this part, we have observed that the temporal whitening step for the multi-temporal stack we considered had a limited impact. We chose to skip this step and compare the performance of our network trained directly on multi-temporal stacks with other reference methods. A study of the impact of temporal correlation and our proposed whitening step are given in section 5.3.5.

Parameters used for our training are recalled in Table 5.1, last column. Figure 5.10 shows two excerpts taken from the TerraSAR-X stacks used for training. Note that, given our self-supervised training strategy, our network can be tested on the same dataset as used for training. When applying the network to other datasets, the performance might drop if the type of area differs significantly (e.g, training on urban areas and testing on mountainous regions) due to a poor generalization. A fine-tuning step on the data of interest using the self-supervised loss is then preferable.

The Figure 5.10 contains two panels with the same numbering, each corresponding to a different stack. The single-look amplitude is shown in (a). In order to identify low-contrasted structures and fine details, the temporal average computed over the whole stack is shown in (b). Due to the changes that occur throughout the time series, this image is not directly comparable to image (a) but is still useful to analyze temporally-stable structures present in the scene given that speckle is strongly reduced by temporal averaging. Areas with fluctuating reflectivities lead to an average value that differs from the reflectivity at the date of interest. Restoration results obtained with several speckle reduction methods are shown in each panel: (c) the mono-date MERLIN network, (d and g) the proposed multi-temporal MERLIN networks, and two baseline patch-based methods: (e and h) MSAR-BM3D introduced in [54] and (f and i) 2SPPB proposed in [15]. Multi-temporal methods are applied to a subset of 4 dates (the reference date + 3 additional dates) in the second row of the figure, or 16 dates (the reference date + 15 additional dates) on the last row. Temporal leakages can be observed in the results of MSAR-BM3D and 2SPPB: spurious information from the other dates contaminate the reference date. This is especially visible by the attenuation of the dark area in the center left of the image on the left pannel. In that respect, multi-temporal MERLIN

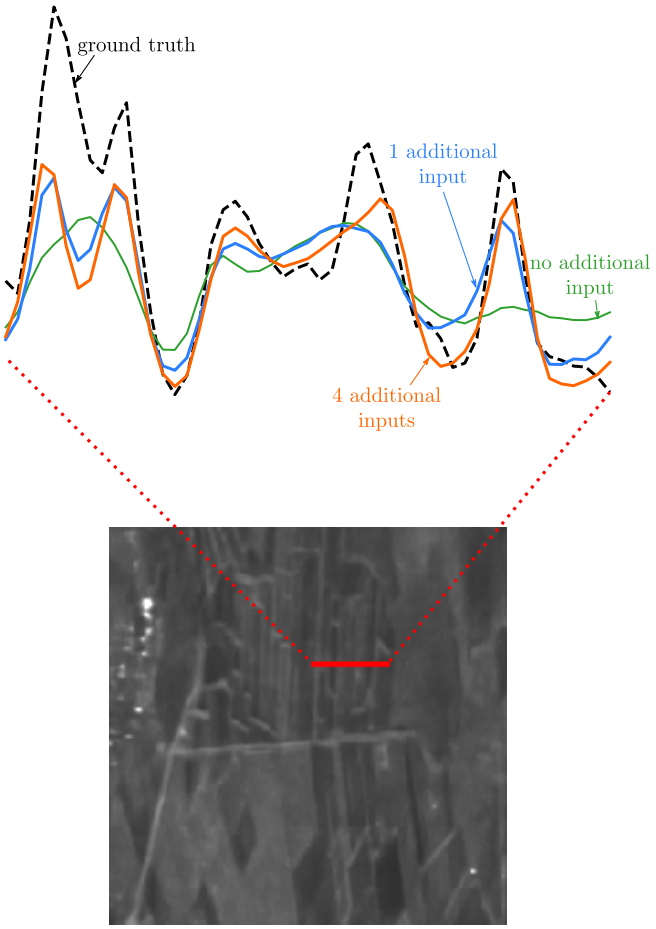


Figure 5.9: Reflectivites profile along the red line, Marais1, date 14. The profile associated to MERLIN network estimation (green line) is blunt, meaning that the edges of the small observed structures are blurred. The more additional inputs there are, the sharper the profile lines, leading to a better retrieving of small structures.

offers much better results with restored reflectivities in good match with the noisy observation shown in Figure 5.10(a). Edges are sharper and low-contrast structures are better preserved in the case with a limited amount of dates (3 additional inputs): Figure 5.10(d-f) left and right panels.

To illustrate that our self-supervised strategy requires only a modest amount of data and that it can be applied to another satellite, we also trained from scratch the networks with a single stack of 12 Sentinel-1 images in Stripmap mode with 2000×2000 pixels. We compare in Figure 5.11 our despeckling results to the images obtained with the same baseline methods as in Figure 5.10 (namely, MSAR-BM3D and 2SPPB). Similar observations can be made: fine structures are better restored by the multi-temporal MERLIN and fewer artifacts can be noticed. Increasing the number of input images systematically leads to an improvement of the output image. Note that the use of

MERLIN loss to train a network on Sentinel-1 in TOPS mode still requires some additional work in order to find the adequate pre-processing that would lead to statistically independent real and imaginary parts [19].

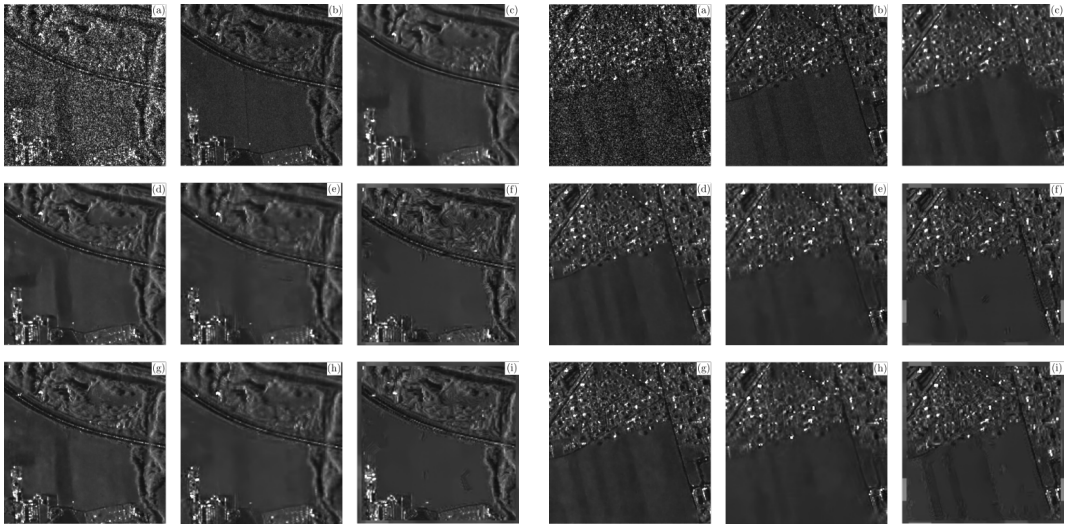


Figure 5.10: Multi-temporal denoising of TerraSAR-X images ©DLR (SLC images with actual speckle): left panel, city of Saint-Gervais (France); right panel, city of Domancy (France). Each panel shows (a) the noisy image; (b) the temporal average of all 26 images of the stack; (c) mono-date MERLIN filtering [19]; (d) multi-temporal MERLIN with 3 additional inputs; (e) MSAR-BM3D [54] with 3 additional inputs, (f) 2SPPB with 3 additional inputs [15]; (g) multi-temporal MERLIN with 15 additional inputs; (h) MSAR-BM3D [54] with 15 additional inputs, (i) 2SPPB with 15 additional inputs [15].

5.3.5 Influence of temporal correlation

Images acquired in interferometric configuration may suffer from correlations along the temporal axis, as discussed in Section 5.3.1. This is not ideal in the context of multi-temporal filtering as it reduces the potential benefit of temporal speckle averaging. Beyond this limitation, we illustrate here that if neglected, i.e., if the proposed temporal decorrelation step is omitted, this type of correlations impacts the despeckling performance of networks trained with the multi-temporal MERLIN loss function. Thus the independence assumption between the inputs and the component used for self-supervision is no longer valid and a whitening step is needed during the pre-processing of the stack.

Experimental results on simulated speckle

We want to highlight the importance of temporal correlations and its impact on the despeckling results. We repeat the experiment with simulated speckle presented in section 5.3.5, this time introducing temporal correlations with a coherence matrix $\mathbf{\Gamma}_k$ identical for all pixels k, and following

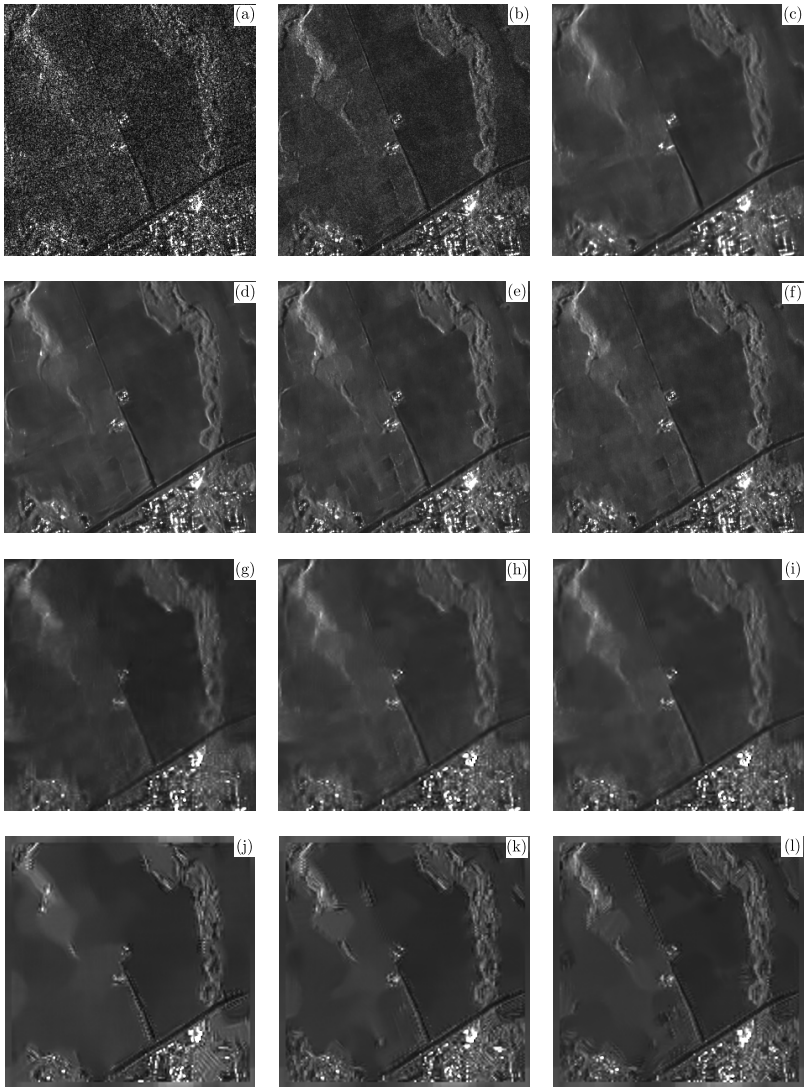


Figure 5.11: Multi-temporal denoising of Sentinel-1 Stripmap images of the Reunion island (France). Each panel shows (a) the noisy image; (b) the temporal average of all 12 images of the stack; (c) mono-date MERLIN filtering [19]; (d) multi-temporal MERLIN with 1 additional input; (e) multi-temporal MERLIN with 3 additional inputs; (f) multi-temporal MERLIN with 7 additional inputs; (g) MSAR-BM3D [54] with 1 additional input; (h) MSAR-BM3D with 3 additional inputs; (i) MSAR-BM3D with 7 additional inputs; (j) 2SPPB [15] with 1 additional input; (k) 2SPPB with 3 additional inputs; (l) 2SPPB with 7 additional inputs.

a simple temporal decorrelation model

$$\forall k,\ \mathbf{\Gamma}_k(t_i, t_j) = \exp\left(-\frac{|t_i - t_j|}{\tau}\right), \tag{5.28}$$

where τ is a characteristic decorrelation time. Rather than reporting how the despeckling performance degrades as a function of parameter τ, we use the more intuitive average coherence $\bar{\gamma}$ defined by

$$\bar{\gamma} = \frac{1}{T^2} \sum_{1 \leq i,j \leq T} \mathbf{\Gamma}_k(t_i, t_j)\,. \tag{5.29}$$

Figure 5.12 reports the evolution of the PSNR of restored images computed on log reflectivities, as a function of the average coherence $\bar{\gamma}$ for a network that uses two additional inputs. Up to $\bar{\gamma} \approx 0.2$ the performance is almost unchanged. Then it degrades significantly. At $\bar{\gamma} \approx 0.45$, the PSNR value is no better than that reached by a network with no additional input. Beyond $\bar{\gamma} \approx 0.45$, it is worse to include additional dates. The reason is that the temporal correlations of speckle lead the network to "cheat" and to partially guess the speckled component in the images used to supervise the training. Once trained, the network systematically leaves a large fraction of the speckle fluctuations unchanged as it expects these fluctuations to also match well the temporally-correlated image used for the supervision.

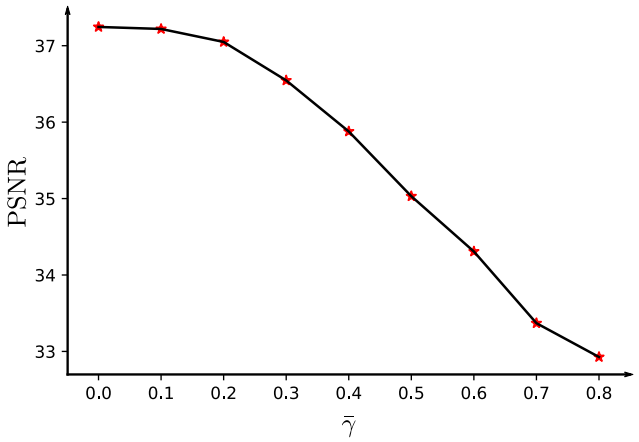


Figure 5.12: Evolution of the restoration performance (PSNR values computed on log reflectivities) as a function of the average coherence $\bar{\gamma}$ of the multi-temporal stack.

Proposed whitening step and its impact

We illustrate our *optional* preprocessing step that performs a temporal decorrelation with respect to the reference date. Our decorrelation is achieved as follows:

1) each SLC image of the stack is decomposed into a dominant scatterers component and a background component;

2) interferograms with respect to the reference date are computed on the background components;

3) at each pixel, a temporal whitening is performed;

4) the contribution of dominant scatterers is reintroduced.

These different steps are illustrated in Figure 5.13.

We perform step 1) with the method described in [7]: the low-pass filtering effect introduced by the SAR system response (step d of the generative model of Figure 5.4) is first compensated by resampling and spectral equalization. Then an *a contrario* framework is applied to detect the cardinal sines of the dominant scatterers. The contribution of the dominant scatterers is then subtracted from the image and the original spectral apodization is reapplied.

In step 2), we estimate interferograms between all pairs of images drawn from the stack of background components $\dot{\boldsymbol{z}} - \dot{\boldsymbol{d}}$. In our experiments, we use the MuLoG algorithm to compute these interferograms. This step is computationally intensive since forming all possible interferograms requires $\mathcal{O}(T^2)$ interferogram estimations for a multi-temporal stack with T dates.

The temporal whitening applied for step 3) is based on the local coherence matrix that has been computed before. Correlations along the temporal axis of the speckle depend both on the coherence matrices $\boldsymbol{\Gamma}_k$ (capturing the temporal decorrelation of the scene); and on the shifts induced by the phases $\boldsymbol{\psi}_t$ (accounting for the geometrical decorrelation due to the change of incidence angles introduced by the interferometric baseline). To reduce these correlations, a whitening process can be designed based on the covariance values:

$$\mathrm{Cov}[\dot{\boldsymbol{z}}(t_{\mathrm{ref}}, k);\, \dot{\boldsymbol{z}}(t_i, k)] = \mathbb{E}[(\dot{\boldsymbol{z}}_{\mathrm{ref}}(k) - \dot{\boldsymbol{d}}_{\mathrm{ref}}(k))(\dot{\boldsymbol{z}}_i(k) - \dot{\boldsymbol{d}}_i(k))^*]$$

The dominant scatterer component $\dot{\boldsymbol{d}}$ can be extracted from the images using an iterative algorithm [7]. The 2×2 interferometric covariance matrices is written

$$\begin{pmatrix} \mathrm{Cov}[\dot{\boldsymbol{z}}(t_{\mathrm{ref}}, k);\, \dot{\boldsymbol{z}}(t_{\mathrm{ref}}, k)] & \mathrm{Cov}[\dot{\boldsymbol{z}}(t_{\mathrm{ref}}, k);\, \dot{\boldsymbol{z}}(t_i, k)] \\ \mathrm{Cov}[\dot{\boldsymbol{z}}(t_i, k);\, \dot{\boldsymbol{z}}(t_{\mathrm{ref}}, k)] & \mathrm{Cov}[\dot{\boldsymbol{z}}(t_i, k);\, \dot{\boldsymbol{z}}(t_i, k)] \end{pmatrix}$$

at each pixel k. It can then be estimated by using an algorithm such as MuLoG [14]. We propose to use these estimations to approximate the $2N \times 2N$ covariance matrix

$$\mathrm{Cov}\left[\begin{pmatrix} \dot{\boldsymbol{z}}_i \\ \dot{\boldsymbol{z}}_{\mathrm{ref}} \end{pmatrix}\right] \approx \begin{pmatrix} \mathbf{D}_{i\,i} & \mathbf{D}_{\mathrm{ref}\,i}^\dagger \\ \mathbf{D}_{\mathrm{ref}\,i} & \mathbf{D}_{\mathrm{ref\,ref}} \end{pmatrix}, \tag{5.30}$$

where the four $N \times N$ blocks are diagonal. Neglecting off-diagonal values of the matrices $\mathbf{D}_{i\,i}$ and $\mathbf{D}_{\mathrm{ref\,ref}}$ amounts to considering a limited spatial correlation length meaning that the SAR impulse response is close to a Dirac. Neglecting off-diagonal values of the matrices $\mathbf{D}_{i\,\mathrm{ref}}$ and $\mathbf{D}_{\mathrm{ref}\,i}$ is justified when the multi-temporal stack is in interferometric configuration: a shift by one or more pixels of the image $\dot{\boldsymbol{z}}_i$ with respect to image $\dot{\boldsymbol{z}}_{\mathrm{ref}}$ drastically reduces the interferometric coherence and the diagonal of $\mathbf{D}_{i\,\mathrm{ref}}$ is dominant.

From the expression of the covariance matrix $\mathrm{Cov}[\tilde{\boldsymbol{z}}]$ given in equation 5.3.1 and the definition of $\dot{\boldsymbol{z}}$ in equation 5.14, we can derive the exact covariances $\mathrm{Cov}[\dot{\boldsymbol{z}}_i]$ and $\mathrm{Cov}[\dot{\boldsymbol{z}}_{\mathrm{ref}}]$ of the centered complex amplitudes of the considered pair of SAR images:

$$\mathrm{Cov}[\dot{\boldsymbol{z}}_i] = \mathbf{Q}\mathrm{diag}(\boldsymbol{r}_i)\mathbf{Q}^\dagger$$
$$\mathrm{Cov}[\dot{\boldsymbol{z}}_{\mathrm{ref}}] = \mathbf{Q}\mathrm{diag}(\boldsymbol{r}_{\mathrm{ref}})\mathbf{Q}^\dagger$$

We approximate this covariance matrix by its diagonal:

$$\mathrm{Cov}[\dot{\boldsymbol{z}}_i] \approx \mathbf{D}_{i\,i}$$
$$\mathrm{Cov}[\dot{\boldsymbol{z}}_{\mathrm{ref}}] \approx \mathbf{D}_{\mathrm{ref\,ref}}$$

with $\tilde{\boldsymbol{r}}_i$ the diagonal of matrix $\mathbf{Q}\mathrm{diag}(\boldsymbol{r}_i)\mathbf{Q}^\dagger$ and $\tilde{\boldsymbol{r}}_{\mathrm{ref}}$ the diagonal of matrix $\mathbf{Q}\mathrm{diag}(\boldsymbol{r}_{\mathrm{ref}})\mathbf{Q}^\dagger$. These vectors correspond to a low-pass filtered version of the reflectivity maps, according to the SAR response $\mathbf{Q}$.
The anti-diagonal blocks are approximated by

$$\mathbf{D}_{\mathrm{ref}\,i} = \mathrm{diag}(\tilde{\gamma}_{i\,\mathrm{ref}}\sqrt{\tilde{\boldsymbol{r}}_i\tilde{\boldsymbol{r}}_{\mathrm{ref}}})$$

where products between vectors are applied entry-wise, and $\tilde{\boldsymbol{\gamma}}_{i\,\mathrm{ref}} \in \mathbb{C}^N$ is the vector of complex-valued coherences between dates t_i and t_{ref} ($\forall k,\, 0 \leq |\tilde{\boldsymbol{\gamma}}_{i\,\mathrm{ref}}(k)| \leq 1$).
The covariance matrix of a pair of complex amplitudes at a pixel k is finally given by:

$$\begin{aligned}\mathrm{Cov}\left[\begin{pmatrix}\dot{\boldsymbol{z}}(t_i,k)-\dot{\boldsymbol{d}}(t_i,k)\\ \dot{\boldsymbol{z}}(t_{\mathrm{ref}},k)-\dot{\boldsymbol{d}}(t_{\mathrm{ref}},k)\end{pmatrix}\right] &= \mathrm{Cov}\left[\begin{pmatrix}\dot{\boldsymbol{z}}(t_i,k)\\ \dot{\boldsymbol{z}}(t_{\mathrm{ref}},k)\end{pmatrix}\right]\\ &= \begin{pmatrix}\tilde{\boldsymbol{r}}(t_i,k) & \tilde{\gamma}_{i\,\mathrm{ref}}(k)\sqrt{\tilde{\boldsymbol{r}}(t_i,k)\tilde{\boldsymbol{r}}(t_{\mathrm{ref}},k)}\\ \tilde{\gamma}^*_{i\,\mathrm{ref}}(k)\sqrt{\tilde{\boldsymbol{r}}(t_i,k)\tilde{\boldsymbol{r}}(t_{\mathrm{ref}},k)} & \tilde{\boldsymbol{r}}(t_{\mathrm{ref}},k)\end{pmatrix}. \end{aligned} \tag{5.31}$$

The covariance along the temporal dimension between the image of reference $\dot{\boldsymbol{z}}_{\mathrm{ref}}$ and the image at date t_i $\dot{\boldsymbol{z}}_i$ modeled by (5.30) can be suppressed by multiplying each $2N$ vector $(\dot{\boldsymbol{z}}_i - \dot{\boldsymbol{d}}_i, \dot{\boldsymbol{z}}_{\mathrm{ref}} - \dot{\boldsymbol{d}}_{\mathrm{ref}})$ by a whitening matrix $\mathbf{W}$, leading to the whitened pair of images $(\mathring{\boldsymbol{z}}_i, \mathring{\boldsymbol{z}}_{\mathrm{ref}})$:

$$\begin{pmatrix}\mathring{\boldsymbol{z}}_i\\ \mathring{\boldsymbol{z}}_{\mathrm{ref}}\end{pmatrix} = \mathbf{W}\begin{pmatrix}\mathring{\boldsymbol{z}}_i - \dot{\boldsymbol{d}}_i\\ \mathring{\boldsymbol{z}}_{\mathrm{ref}} - \dot{\boldsymbol{d}}_{\mathrm{ref}}\end{pmatrix} + \begin{pmatrix}\dot{\boldsymbol{d}}_i\\ \dot{\boldsymbol{d}}_{\mathrm{ref}}\end{pmatrix} \tag{5.32}$$

with

$$\mathbf{W} = \mathbf{\Pi}^{-1}\begin{pmatrix}\mathbf{W}_1^\dagger & & \mathbf{0}\\ & \ddots & \\ \mathbf{0} & & \mathbf{W}_N^\dagger\end{pmatrix}\mathbf{\Pi} \tag{5.33}$$

and

$$\mathbf{W}_k\mathbf{W}_k^\dagger = \mathrm{Cov}\left[\begin{pmatrix}\dot{\boldsymbol{z}}(t_i,k)-\dot{\boldsymbol{d}}(t_i,k)\\ \dot{\boldsymbol{z}}(t_{\mathrm{ref}},k)-\dot{\boldsymbol{d}}(t_{\mathrm{ref}},k)\end{pmatrix}\right]^{-1}. \tag{5.34}$$

The matrices $\mathbf{W}_k$ can be obtained by Cholesky factorization of the inverse of the covariance matrix given in equation (5.34).
The closed-form expression of the Cholesky factorization in equation (5.34) leads to a simple definition of the whitened pair

$$\begin{cases}\mathring{\boldsymbol{z}}(t_i,k) = \tau\dot{\boldsymbol{z}}(t_i,k) + (1-\tau)\dot{\boldsymbol{d}}(t_i,k) - \sqrt{\frac{\tilde{\boldsymbol{r}}(t_i,k)}{\tilde{\boldsymbol{r}}(t_{\mathrm{ref}},k)}}\tau\tilde{\gamma}^*_{i\,\mathrm{ref}}(k)(\dot{\boldsymbol{z}}(t_{\mathrm{ref}},k) - \dot{\boldsymbol{d}}(t_{\mathrm{ref}},k))\\ \mathring{\boldsymbol{z}}(t_{\mathrm{ref}},k) = \dot{\boldsymbol{z}}(t_{\mathrm{ref}},k)\,,\end{cases} \tag{5.35}$$

where $\tau = 1/\sqrt{1-|\tilde{\gamma}_{i\,\mathrm{ref}}(k)|^2}$.

Note that only the complex amplitude $\mathring{\boldsymbol{z}}_{t_i}$ is modified while $\mathring{\boldsymbol{z}}_{\text{ref}}$ is left unchanged. This whitening procedure can thus be repeated for all pairs (t_i, t_{ref}), with $1 \leq t_i \leq T$ and $t_i \neq t_{\text{ref}}$, thereby producing a pre-processed stack in which images are all decorrelated with respect to the reference date t_{ref} and the decorrelated images provide information for the self-supervised training. Only the statistical independence with respect to this target date matters for the validity of the self-supervision used in section 5.3.3.

We also need to prove that the whitened pair $(\mathring{\boldsymbol{z}}(t_i, k), \mathring{\boldsymbol{z}}(t_{\text{ref}}, k))$ has indeed a diagonal covariance matrix to perform the training.

We can rewrite the whitened pair as follows:

$$\begin{pmatrix} \mathring{\boldsymbol{z}}(t_i, k) \\ \mathring{\boldsymbol{z}}(t_{\text{ref}}, k) \end{pmatrix} = \begin{pmatrix} \tau & -\sqrt{\frac{\tilde{\boldsymbol{r}}(t_i,k)}{\tilde{\boldsymbol{r}}(t_{\text{ref}},k)}}\tau\tilde{\gamma}^*_{i\,\text{ref}}(k) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{\boldsymbol{z}}(t_i, k) - \dot{\boldsymbol{d}}(t_i, k) \\ \dot{\boldsymbol{z}}(t_{\text{ref}}, k) - \dot{\boldsymbol{d}}(t_{\text{ref}}, k) \end{pmatrix} + \begin{pmatrix} \dot{\boldsymbol{d}}(t_i, k) \\ \dot{\boldsymbol{d}}(t_{\text{ref}}, k) \end{pmatrix}. \tag{5.36}$$

Since the centered dominant component is deterministic, it follows from equations (5.36) and (5.31) that

$$\begin{aligned} \text{Cov}\left[\begin{pmatrix} \mathring{\boldsymbol{z}}(t_i, k) \\ \mathring{\boldsymbol{z}}(t_{\text{ref}}, k) \end{pmatrix}\right] &= \text{Cov}\left[\begin{pmatrix} \mathring{\boldsymbol{z}}(t_i, k) - \dot{\boldsymbol{d}}(t_i, k) \\ \mathring{\boldsymbol{z}}(t_{\text{ref}}, k) - \dot{\boldsymbol{d}}(t_{\text{ref}}, k) \end{pmatrix}\right] \\ &= \begin{pmatrix} \tau & -\sqrt{\frac{\tilde{\boldsymbol{r}}(t_i,k)}{\tilde{\boldsymbol{r}}(t_{\text{ref}},k)}}\tau\tilde{\gamma}^*_{i\,\text{ref}}(k) \\ 0 & 1 \end{pmatrix} \text{Cov}\left[\begin{pmatrix} \dot{\boldsymbol{z}}(t_i, k) \\ \dot{\boldsymbol{z}}(t_{\text{ref}}, k) \end{pmatrix}\right] \begin{pmatrix} \tau & 0 \\ -\sqrt{\frac{\tilde{\boldsymbol{r}}(t_i,k)}{\tilde{\boldsymbol{r}}(t_{\text{ref}},k)}}\tau\tilde{\gamma}_{i\,\text{ref}}(k) & 1 \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\boldsymbol{r}}(t_i, k) & 0 \\ 0 & \tilde{\boldsymbol{r}}(t_{\text{ref}}, k) \end{pmatrix}. \end{aligned} \tag{5.37}$$

This proves that, for each pixel k, the two complex amplitudes are decorrelated. Since they are jointly Gaussian and decorrelated, they are statistically independent.

Step 3) is thus very fast since it only requires applying pixelwise the simple whitening transform of equation (5.35).

Finally, the reintroduction of dominant scatterers in step 4) leads to the temporally whitened stack $\mathring{\boldsymbol{z}}$.

In order to assess the impact of this temporal decorrelation step, we compared the performance of the same network trained in one case directly on a stack of 26 TerraSAR-X images $\dot{\boldsymbol{z}}$ (i.e., the spectra have been shifted to center the spectrum of the reference date, but no temporal decorrelation step has been carried out). And in the other case, using a pre-processed stack with our spectrum centering plus the temporal decorrelation technique. Coherences between the first two images of the original and the pre-processed stacks computed with the MuLoG algorithm are presented in Figure 5.14. It shows that the proposed whitening step strongly reduces the coherence. Despeckling results are presented in Figure 5.15 and very few differences can be observed even though slight changes may be noted around some scatterers. The average coherence on this stack of TerraSAR-X images is equal to 0.23, which corresponds to a mild level of correlation with a negligible impact on the despeckling performance, as shown in our experiments with simulated speckle reported in Figure 5.12. This illustrates that, even in the case of a satellite with interferometric capabilities, the computationally heavy preprocessing step of temporal decorrelation can be skipped when the coherence level is moderate.

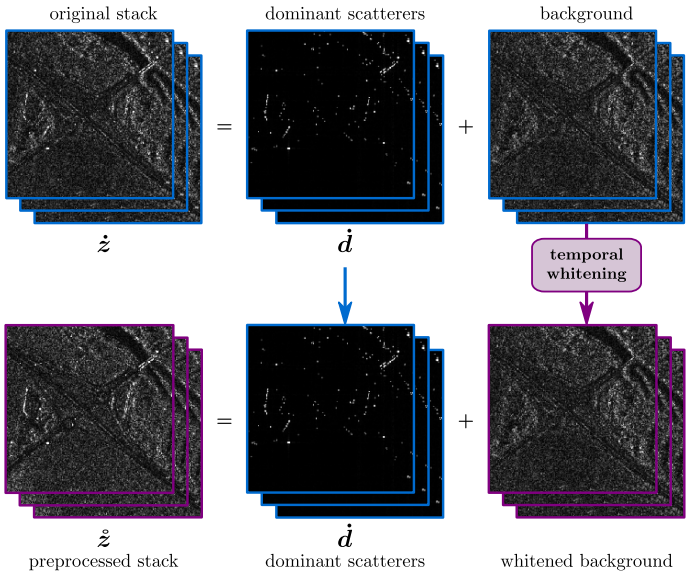


Figure 5.13: Illustration of the preprocessing step to reduce temporal correlations of the speckle: (first row) the multi-temporal stack is decomposed into dominant scatterers and background using the method in [7]; (second row) the background component is then whitened and the dominant scatterers are added back to produce the preprocessed stack.

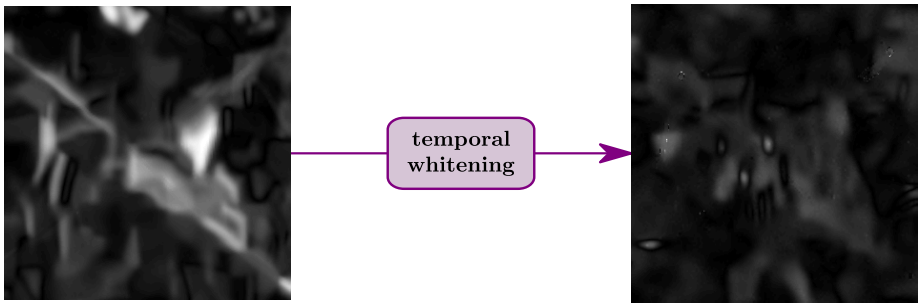


Figure 5.14: Coherences computed with the MuLoG algorithm on the 2 first dates (2009/05/31 and 2009/06/11) of the Domancy TerraSAR-X stack ©DLR. The studied area is introduced in 5.13; left: estimated coherence before the whitening step; right: estimated coherence after the whitening step.

5.4 Conclusion

Multi-temporal despeckling have been explored in this chapter. First, methods based on temporal averaging or the use of a super-image avec been introduced in section 5.2

A generative model based on the decomposition of the SLC images into a speckle component and a dominant scatterers component has been developed in section 5.3.1. It breaks down the different sources of statistical correlation between spatial, temporal, and real/imaginary components of the

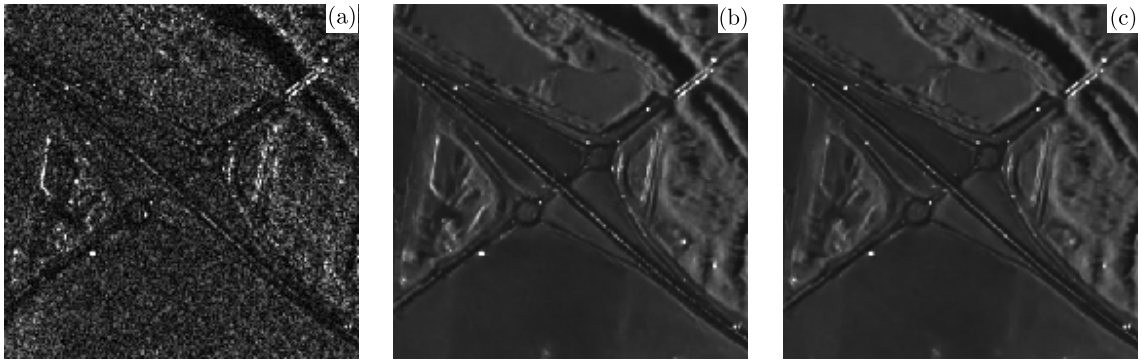


Figure 5.15: Impact of the temporal whitening step on multi-temporal denoising of TerraSAR-X images ©DLR, city of Domancy (France). (a) noisy image; (b) multi-temporal MERLIN with 2 additional inputs trained on one TerraSAR-X stack without the temporal whitening step; (c) multi-temporal MERLIN with 2 additional inputs trained on one TerraSAR-X stack with the temporal whitening step.

complex amplitudes of SAR images. In sections 5.3.4 and 5.3.5, we have shown that under some assumptions like a low coherence or an adequate preprocessing step (temporal whitening), the self-supervised training strategy MERLIN can be extended to stacks of SLC images.
This strategy improves the despeckling performance achieved by mono-date networks by exploiting temporal redundancies of the scene and temporal fluctuations of speckle. Our quantitative analysis shows an improvement of the restored reflectivities, a refined spatial resolution, and very few temporal contamination by possible changes in the additional dates provided in input. The networks trained directly on SAR images, without groundtruth, produce restored images of higher quality compared to state-of-the-art techniques.

Chapter 6

Estimating the uncertainties associated to the restored images

This chapter presents our work on estimating uncertainties for deep learning-based despeckling methods.

The lack of ground truth image, as stated in the first chapter, has made the quantitative evaluation of despeckling methods quite difficult. Even though metrics exist, such as the equivalent number of looks computed on homogeneous areas of the restored image, or the analysis of ratio images [80, 81] to form the equivalent of method noise in [25], none is consensual.
With deep learning methods, it is always difficult to find clues that the network has created information, which is a known problem. The despeckling of bright scatterers is tricky as it is even difficult to differentiate them from speckle in the noisy image most of the time.
We want to provide both a restored image and an uncertainty map. We will then be able to locate areas where the network is not confident and the despeckling could not be blindly trusted.

This chapter provides an introduction and a general overview about uncertainty estimation in deep learning, and then presents two kinds of methods that were developed: the first one is related to the estimation of distribution parameters for each pixel of the image, which is statistically well grounded but proved very challenging to train in a self-supervised fashion; the second one is based on the estimation of the expected difference map between reflectivities predicted from the real and imaginary part of an SLC image using the MERLIN framework [19] and is much better.

6.1 Overview of uncertainty quantification in machine learning

Basic neural networks do not estimate uncertainty. Most of the time, uncertainties are due to undesired fluctuations in the data (this is what we call *data uncertainty*) and are linked to a lack of knowledge of the neural network (this is what we call *model uncertainty*). It is quite difficult to perfectly unmix data and model uncertainties. This section will provide an overview of uncertainty, starting from its definition and where it comes from, to the different kinds of methods that have been proposed these recent years.

6.1.1 What is uncertainty?

When we are training a network and estimating uncertainty, the whole process can be decomposed into 4 different steps starting from the raw information used in the training and ending to a prediction by a neural network with quantified uncertainties: the data acquisition process, the design and training of the network, the inference phase and the prediction of uncertainty. During each one of these steps, uncertainty can be introduced and can come from the variability in real world situations (diversity in the test data set different from the one in the training set, a phenomenon known as distribution shift), error and noise in the labels or ground truth (referenced as label noise), errors in the model structure (for example, deep networks tend to be overconfident due to overfitting), errors in the training procedure (coming from the choice of the hyperparameters and the stochastic process used for gradient estimation during the training).

We are mostly interested in the uncertainty propagated onto a prediction $\hat{\boldsymbol{r}}$.
The predictive uncertainty associated with $\hat{\boldsymbol{r}}$ is in general separated into *data uncertainty* (that directly stems from the data) and *model uncertainty* (caused by shortcomings in the model such as errors in the training procedure, insufficient model structure, lack of knowledge due to unknown data and bad coverage of the training data set). While model uncertainty can be theoretically reduced by improving the architecture, the learning process or the training set, the data uncertainty cannot be explained away.

Evaluating the quality of the uncertainty estimates is a challenging task: the quality of the uncertainty estimation depends on the underlying method for estimating uncertainty. There is a lack of ground truth uncertainty estimates and defining ground truth uncertainty estimates is challenging. No metric is consensual, meaning that uncertainty is defined differently in different machine learning tasks: prediction intervals or standard deviations are used in regression tasks while entropy is used in classification or segmentation tasks.

The uncertainty can be measured as the width of an interval. The Prediction Interval Coverage Probability represents the percentage of test predictions that fall into a prediction interval and is defined as

$$\textbf{PICP} = \frac{c}{n} \tag{6.1}$$

where n is the total number of predictions and c the number of ground truth values that are actually captured by the predicted intervals. The larger the interval, the better the PICP which can be a undesired behavior: we do not want to predict very large intervals.

Calibration of the uncertainty is also important. A predictor is said to be well-calibrated if the derived predictive confidence represents a good approximation of the actual probability of correctness. For regression task, the calibration can be defined such that predicted confidence intervals should match the confidence intervals empirically computed from the data set. Calibration can be performed during the training phase by modifying the loss function in order to have a calibrated network or post-processing methods applied after the training step to adjust the predictions afterwards.

6.1.2 Different methods to estimate uncertainties

Based on [82], there are 4 different types of methods to estimate uncertainties:

- **Test-time augmentation methods**: they give the prediction based on a single deterministic network but augment the input data at test-time.

- **Ensemble methods**: they combine the predictions of several different deterministic networks at inference.
- **Single deterministic methods**: they give a prediction based on one single forward pass within a deterministic network. The uncertainty quantification is either derived by using additional methods or is directly predicted by the network.
- **Bayesian methods**: they cover all kinds of stochastic networks, and focus on model uncertainty estimation.

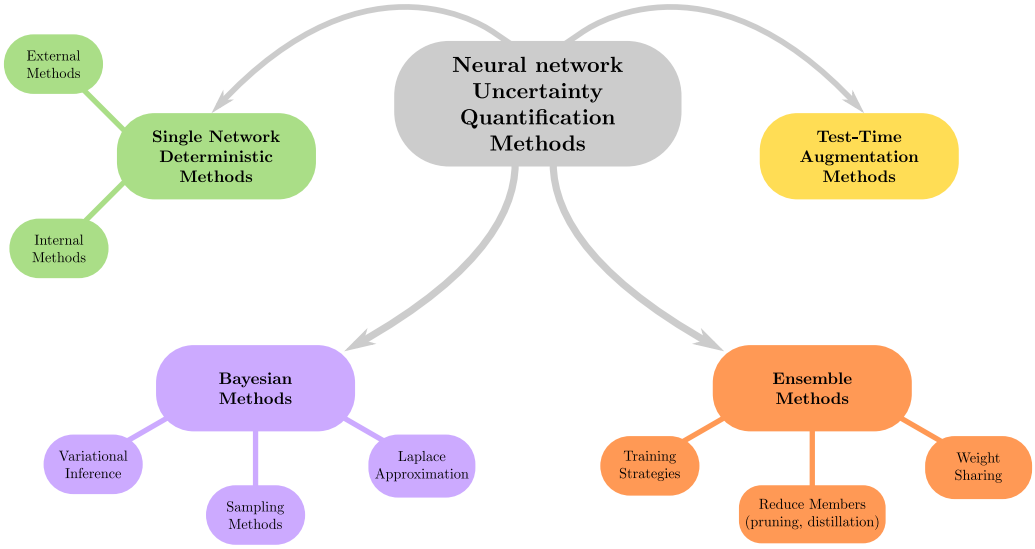


Figure 6.1: This figure sums up the 4 types of methods to compute uncertainty. Some of then are described in the following paragraphs. Illustration inspired by one of the figures in [82].

Test-Time data augmentation methods

Test-Time augmentation methods enable uncertainty quantification thanks to data augmentation. Different predictions are done by the network of interest based on different augmentations of the original input sample, and uncertainty is computed based on these multiple predictions. Possible augmentation policies are described in [83].

Ensemble Methods

Ensemble methods rely on the estimation of uncertainty using multiple *members* meaning different variations of one algorithm. The main argument of these approaches is that a group of decision makers tend to make better decisions than a single decision maker. This could be implemented by averaging over the members' predictions for example. The accuracy is improved and it gives an intuitive way of representing model uncertainty on a prediction by evaluating the variability among the members' predictions.

Indeed, each network used for uncertainty estimation is expected to converge to a slightly different local optima during training. To maximize the variety in the behavior among the networks, different approaches can be applied such as random initialization and data shuffle, bagging and boosting, data augmentation and ensemble of different network architecture. The work of Lakshminarayanan et al. [84] is often referenced as a pioneer work in estimation of uncertainty in deep learning using ensemble methods.
A family of ensemble methods used in the literature are know as *Sub-ensembles methods*. They tend to reduce the computational cost by dividing a neural network architecture into two sub-networks. The trunk network is used for the extraction of general information from the input data and the task network is using this information to perform the required task. The weights of each member's trunk are fixed based on the resulting parameters of one single model while the parameters of each ensemble members' task network are trained independently [85, 86, 87].
Even if ensemble methods provide a simple approach for the quantification of uncertainty, they are computationally greedy and will not be studied in details in this work.

Single deterministic methods

The main advantage of single deterministic methods is that they require only a single pass through the deterministic model to obtain an estimation of the uncertainty associated to a prediction $\hat{\boldsymbol{r}}$. Two types of single deterministic approaches can be cited: the ones where a single network is explicitly modeled and trained in order to quantify uncertainties; and the ones that use additional components in order to give an uncertainty estimate on the prediction of the network.
Some methods known as *external uncertainty* quantification approaches, consist in a two-steps framework where the prediction is firstly done and then the uncertainty is estimated based on the later prediction [88, 89]. The training of two neural networks is necessary: one is specific to the prediction task and a second one to uncertainty quantification based on the predictions of the first one. Single deterministic methods can be applied to several networks as a post-processing step. The uncertainty quantification is always based on the method itself and takes into account uncertainty due to the architecture, the training procedure and the training data.

The Bayesian methods

In Bayesian modeling, the model uncertainty is formalized as a probability distribution over the model parameters θ of the network, and the data uncertainty is formalized as a probability distribution over the model outputs $\hat{\boldsymbol{r}}$ given a parameterized model f_θ.
Let D be the training set, the model uncertainty is captured through $p(\theta|D)$ the posterior distribution on the model parameters. Ensemble approaches approximate the posterior distribution by learning several different parameter settings and averaging over the resulting models where Bayesian inference reformulates the problem using the Bayes Theorem:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$$

where $p(\theta)$ is the prior distribution on the model parameters; $p(D|\theta)$ represents the likelihood that the data D is a realization of the distribution predicted by the model parameterized with θ.
Bayesian inference can be conducted through

- Variational inference which approximates the intractable posterior distribution by optimizing over a family of tractable parametric distributions.
- Sampling approaches based on Markov Monte Carlo Sampling and further extensions, where uncertainty can be represented without a parametric model.
- Laplace approximation which simplifies the target distribution by approximating the posterior distribution using the second order Taylor series expansion around the MAP estimate. The computation of the Hessian is required and could be challenging and time-consuming.

When the size of the data set and the number of parameters of the networks are huge, Bayesian approaches are really challenging and time-consuming.

In the following work, we focus on single deterministic methods because a network is trained to predict the restored image and the uncertainty map simultaneously, and external uncertainty is at the center of the estimations in both sections 6.2.1 and 6.2.2. Bayesian approaches will not be used to predict model uncertainty, but the framework described in Section 6.2.1 is inspired by the Bayesian framework.

6.2 From despeckling to uncertainty quantification

This section describes the contributions of this thesis on uncertainty quantification for despeckling of SAR images.

In the first approach, the network is trained to predict the parameters of the chosen distribution for the reflectivity at each pixel of the image. The methodological framework of this approach is engaging, but the experimental results obtained so far are not satisfactory. We can explain this observation by computing the relative error of the uncertainty parameters in a simplified test case of an additive Gaussian noise. This error is too high to expect good results with our network. The second approach is more practical and exploit the MERLIN framework. We can then train a network to predict the expected value of the L2 norm of the difference between the two predicted log-reflectivities using the real and the imaginary part of the image.

In order to have quantitative results, the network is trained on simulated speckle. The ground truth images are obtained again by using the RABASAR extension [67]. The training hyperparameters and further information on the training set are given in Table 6.1.

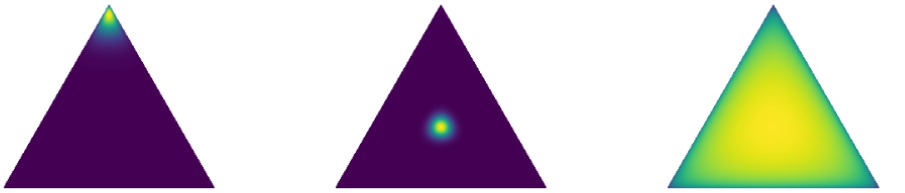
# images	237
patch size	256×256
batch size	8
# patches	1616
# batches	202
# epochs	1000
learning rate	10^{-3} 10^{-4} after 10 epochs 10^{-5} after 910 epochs

Table 6.1: Training parameters for uncertainty estimation using the MERLIN network on simulated speckle.

6.2.1 Prediction of the reflectivity distribution at each pixel

The work [90] proposes a new framework for modeling predictive uncertainty called *Prior Networks* which explicitly models uncertainty triggered by distributional mismatch between the test and training data distributions. The framework is proposed for a classification task and they aim at identifying out-of-distribution samples and detecting misclassification.

Prior Networks explicitly predict parameters of a distribution over a simplex representing the classification space. When the network is confident in its prediction, it should yield a sharp distribution centered on one of the corners of the simplex. When the input is in a data region with noise or class overlap which corresponds to data uncertainty, the network should yield a sharp distribution at the center of the simplex meaning that the network is confident that the input data are not out-of-distribution but the class can not be easily chosen. For out-of-distribution inputs, the network should yield a flat distribution over the simplex meaning that the input data is not understood and thus the prediction is uniformly distributed over the simplex. These various predictions are illustrated in Figure 6.2



(a) Confident Prediction (b) High data uncertainty (c) Out-of-distribution

Figure 6.2: Illustration of the different use cases of the prediction of the Prior Network introduced in [90]. This 2D simplification shows a simplex for a three classes classification task. (a) Confident prediction: sharp and on one vertex (representing one class) of the simplex; (b) High data uncertainty: sharp and at the center because the network is sure it is difficult to classify; (c) Out-of-distribution: wide and at the center, the data distribution is unknown to the network. Figure extracted from [90].

These networks were firstly developed for classification while we are interested in performing despeckling (i.e. regression models). Our approach is inspired by [90]: parameters of a distribution of the reflectivity are predicted by the network at each pixel. The sharper the distribution, the more certain the network is of its prediction.
We still consider in the following the MERLIN mono-date approach with the following loss:

$$\begin{aligned}\mathcal{L}(\boldsymbol{r}(\boldsymbol{a}),\boldsymbol{b}) &= -\log \mathrm{p}(\boldsymbol{b}|\boldsymbol{r}(\boldsymbol{a})) \\ &\approx -\sum_k \log \mathrm{p}(b_k|r_k(\boldsymbol{a})) \text{ if the spatial correlations of the speckle are weak}\end{aligned} \tag{6.2}$$

where $\boldsymbol{b}$ is the imaginary part (another loss can be defined by swapping $\boldsymbol{b}$ with $\boldsymbol{a}$) of the SLC image and $\boldsymbol{r}$ is the sought reflectivity image.
The networks outputs one estimation of the reflectivity for each pixel. However, we want to quantify

the quality and the confidence of the network's predictions. We typically expect the network not to be confident on dominant scaterrers or thin structures.
Instead of predicting one value for each pixel, we want to predict a distribution for each pixel of the reflectivity image: a sharp distribution around a value (the *mode* or the *mean* value depending on the distribution) could be associated with a confidence in the prediction: while a flat distribution is associated with a lack of confidence for this prediction because the network can not make a choice. We define our new loss as follows:

$$\mathcal{L}(\mathrm{p}, \boldsymbol{a}, \boldsymbol{b}) = -\sum_k \log\left[\int_{\mathbb{R}^+} \mathrm{p}(b_k|r_k(\boldsymbol{a}))\ \mathrm{p}(r_k(\boldsymbol{a}))\mathrm{d}r_k(\boldsymbol{a})\right] \tag{6.3}$$

We want to find a parametrized distribution in every pixel k of $\mathrm{p}(r_k(\boldsymbol{a}))$ such that the integral in equation 6.3 is consistent. We will study two distributions for p: the first one is uniform with a parametrization in log-reflectivity; the second one is inverse gamma with a parametrization in reflectivity.
First we will work with log-intensities which is sometimes more convenient. In this case, the following change of variables is needed:

$$\begin{aligned} y &= \log(b_k)^2 \\ x &= \log r_k(\boldsymbol{a}) \end{aligned} \tag{6.4}$$

We need a closed-form expression of the distribution followed by y. Based on the statistics of the speckle introduced in 2.2, the imaginary part $\boldsymbol{b}$ (and of the real part $\boldsymbol{a}$ as well) is normally distributed. We can derive the probability density function $f_{|x}$ of the log of the squared imaginary part $y = \log(b_k)^2$ conditionally to x:

$$\begin{aligned} f_{|x} \colon \mathbb{R}^* &\to \mathbb{R}^{+*} \\ y &\mapsto \frac{e^{-e^{y-x}}}{\sqrt{\pi e^{-y+x}}}. \end{aligned} \tag{6.5}$$

The final expression of our extended Loss in log-scale in pixel k is

$$\ell(\mathrm{p}, y) = -\log\left[\int_{x\in\mathbb{R}} \frac{e^{-e^{y-x}}}{\sqrt{e^{-y+x}}\sqrt{\pi}}\ \mathrm{p}(x)\mathrm{d}x\right] \tag{6.6}$$

The expression of $\mathrm{p}(r_k)$ in equation 6.3, or $\mathrm{p}(x)$ in equation 6.6 is important because it will enable us to interpret the results: if we have a mono modal distribution, we will model an uncertainty which is not necessarily symmetric, and the predicted value can be defined as the mean or the mode of the distribution. If we have a multi-modal distribution, we allow the network to hesitate between several categories of prediction ("field" versus "thin road" for example), but then we would have more parameters to estimate and it would be more challenging. Our first choice for p was the Gaussian distribution, but the integral in equation 6.6 is intractable.

Two distributions over the predicted reflectivities will be considered in the following: the **uniform distribution** over the log reflectivity (equation 6.6); and the **inverse gamma distribution** over the reflectivity (equation 6.3)

Uniform distribution over the log-reflectivities for uncertainty quantification

This section gathers some works on estimating uncertainty based on prediction intervals and describes the proposed approach to estimate uncertainty for despeckling.

The work [91] considers the generation of prediction intervals by neural network for quantifying uncertainty in regression tasks. The high-quality prediction intervals should be as narrow as possible while capturing a specified portion of data. The proposed loss function is based on this property and does not require any distributional assumption. Where neural networks estimate one value for each output point (or pixel if working with images), in most of the cases, prediction intervals directly communicate uncertainty by offering a lower and upper bounds for a prediction and assurance that the realized data point will fall between these bounds with a high probability. In [91], the prediction intervals are estimated by minimizing the Mean Prediction Interval Width (MPIW) with a fixed value of the Prediction Interval Coverage Probability (see equation 6.1). The wider the interval, the more uncertain the network is.

The approach presented in [92] works with any machine learning model, such as neural networks, regardless of the true unknown data distribution or choice of model. They rigorously quantify the uncertainty in an image-valued point prediction. They want to model per-pixel uncertainty intervals of the predicted image that contain the true pixel values with a user-specified probability. The user selects a risk level $\alpha \in (0,1)$ and an error level $\delta \in (0,1)$, then they construct the intervals that contain at least $1-\alpha$ of the ground-truth pixel values with probability $1-\delta$. This algorithm is modular, allowing the user to use the most complex methods to have an estimation of the ground-truth image, including neural networks, all while having uncertainty intervals that reliably render quality of the predictions. The intervals are predicted by a two-steps method: first the lower and upper bounds are estimated using any method such as a neural network, and then calibration of the uncertainty is done by scaling the bounds until they contain the right fraction of the ground truth pixels.

Using the framework introduced at the beginning of this section, we want to estimate prediction intervals when despeckling a SAR image with the MERLIN network. If we suppose that the log reflectivity image x is uniformly distributed on an interval $[x_{\min}, x_{\max}]$ for each pixel of the image, then the network needs to predict the estimated reflectivity and the lower and upper bounds for each pixel. We can deduce an uncertainty map by displaying the width of the intervals equal to $x_{\max} - x_{\min}$.

Based on the equation 6.6, our new loss function is defined as

$$\begin{aligned} \ell(\mathcal{U}_{[x_{\min},x_{\max}]}, y) &= -\log\left[\int_{x\in R} \frac{e^{-e^{y-x}}}{\sqrt{e^{-y+x}}\sqrt{\pi}} \frac{1}{x_{\max}-x_{\min}} \mathrm{d}x\right] \\ &= -\log\left[\frac{\mathrm{erf}(e^{\frac{1}{2}(y-x_{\min})}) - \mathrm{erf}(e^{\frac{1}{2}(y-x_{\max})})}{x_{\max}-x_{\min}}\right] \end{aligned} \tag{6.7}$$

where the erf function is the Gauss error function defined as

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} \mathrm{d}t$$

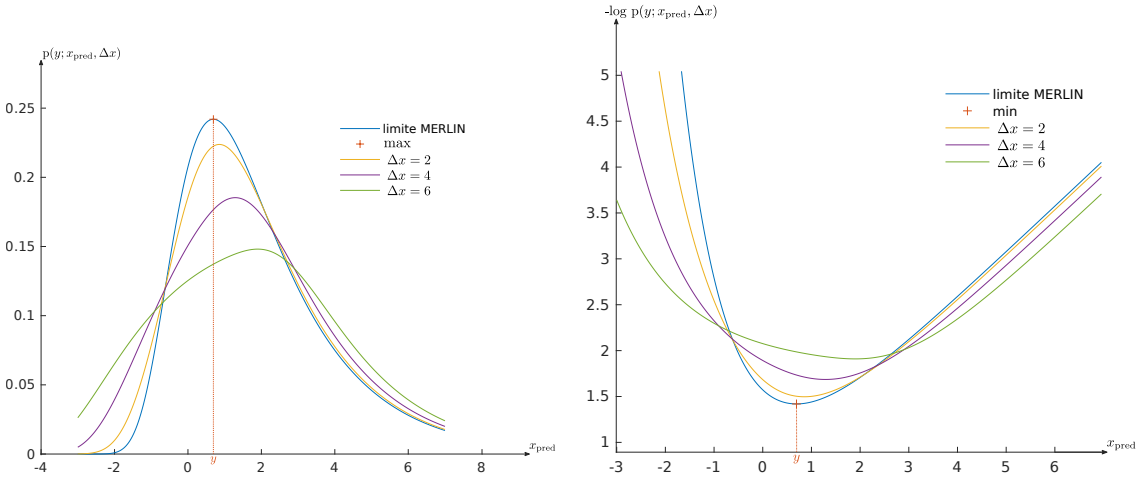
As explained before, calibration is important for uncertainty estimation. An easier way to solve the problem for the network will be to predict the estimated reflectivity x_{pred} and the half-width of the interval $\Delta x = (x_{\max} - x_{\min})/2$ such that:

$$\begin{aligned} x_{\min} &= x_{\text{pred}} - \Delta x \\ x_{\max} &= x_{\text{pred}} + \Delta x \end{aligned} \tag{6.8}$$

The loss function of equation 6.7 can thus be written

$$\ell(\mathcal{U}_{[x_{\text{pred}}-\Delta x, x_{\text{pred}}+\Delta x]}, y) = -\log \underbrace{\left[\frac{\text{erf}(e^{\frac{1}{2}(y-x_{\text{pred}}+\Delta x)}) - \text{erf}(e^{\frac{1}{2}(y-x_{\text{pred}}-\Delta x)})}{2\Delta x} \right]}_{\text{p}(y; x_{\text{pred}}, \Delta x)} \tag{6.9}$$

Optimizing this loss function in 6.9 means that the prediction x_{pred} is in the middle of the interval $[x_{\min}, x_{\max}]$. This assumption makes the problem easier but leaves a smaller degree of freedom. During the training, the output of the network corresponding to Δx needs to be positive, this is enforced by applying a Relu function.
Depending on the width Δx of the predicted intervals, we plot the distribution $\text{p}(y; x_{\text{pred}}, \Delta x)$ and the loss function in Figure 6.3.



Probability density function:
$\text{p}(y; x_{\text{pred}}, \Delta x)$,
as a function of x_{pred},
for a fixed value of y and Δx

Loss function:
$-\log \text{p}(y; x_{\text{pred}}, \Delta x)$
as a function of x_{pred},
for a fixed value of y and Δx

Figure 6.3: Density functions and loss functions (see equation 6.9) for a fixed value of y as a function of x_{pred}. The color of the curves are associated with different values of Δx. We can see that when the parameter Δx is large, the distribution becomes less and less sharp and the Loss minimum value is larger. The ideal case would be a prediction interval of width 0 centered on the right reflectivity value, for our expression. The limit when $x_{\min}$ tends to $x_{\max}$ is the MERLIN Loss. A larger interval means that the network is less confident on its prediction.

Experimental results are given in Figure 6.4: we started by training one network to predict x_{pred} and Δx simultaneously. Unfortunately, the quality of the restored images was much worse

compared to the image obtained using the original MERLIN method. The lower bound $x_{\min}$ and the upper bound $x_{\max}$ of our prediction intervals are very blurred and lack details. We would expect high values of the interval's width around bright scatterers and smaller ones on homogeneous areas. To make the problem easier, we decided to fix the predicted reflectivity during the training step, using the estimation of the original MERLIN network. We then want to predict the Δx value for each pixel of the image. Fewer parameters are to be estimated, and we could suppose that the optimization problem is then made easier. However, experimental results presented in 6.5 show that the estimation of $x_{\min}$ and $x_{\max}$ is still very noisy. Numerous training attempts have been made, and hyperparameters have been tuned while keeping the same architecture as the one introduced in chapter 3.

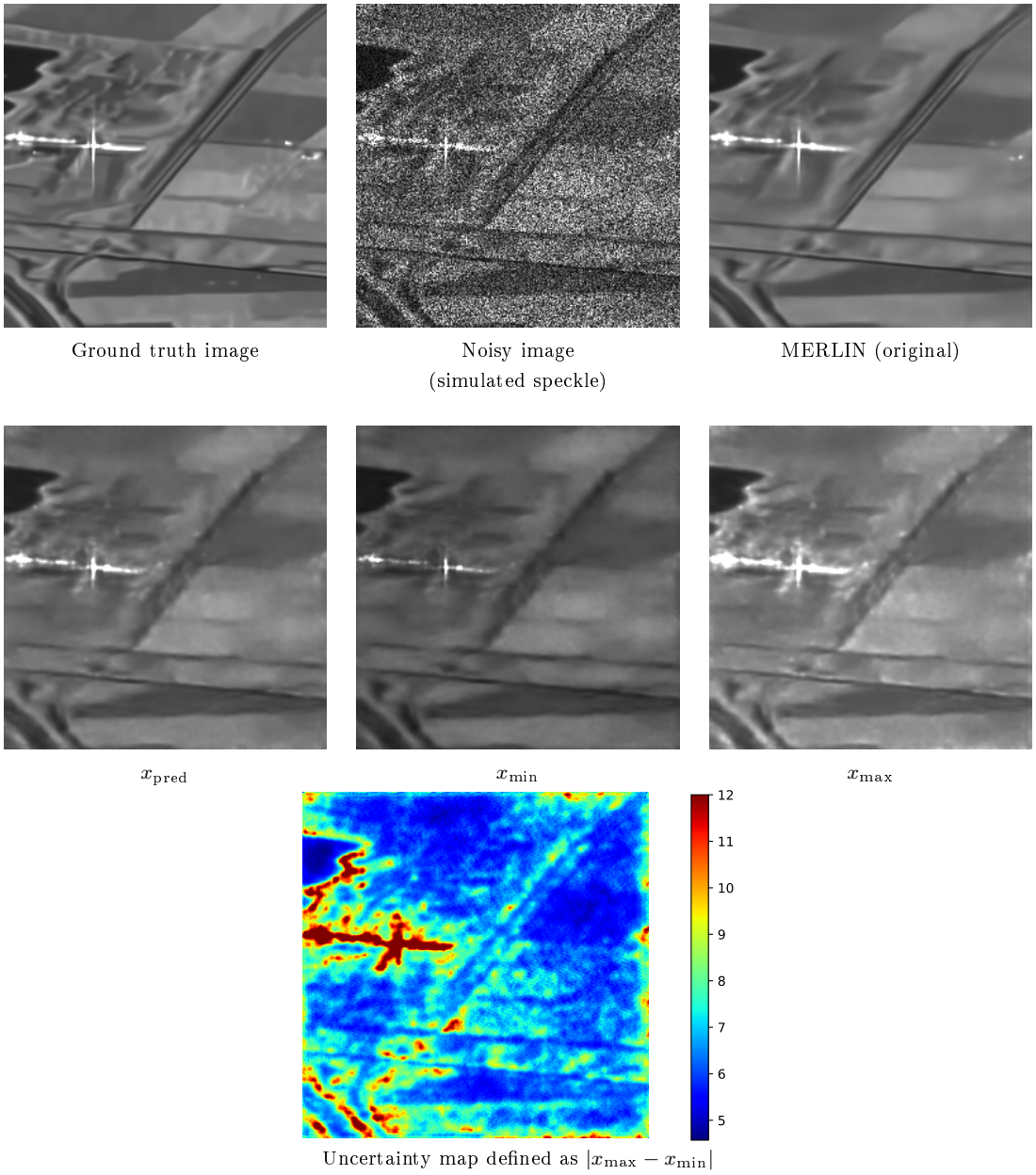


Figure 6.4: Predictions of the network when trained to optimize the loss function corresponding to equation 6.9.

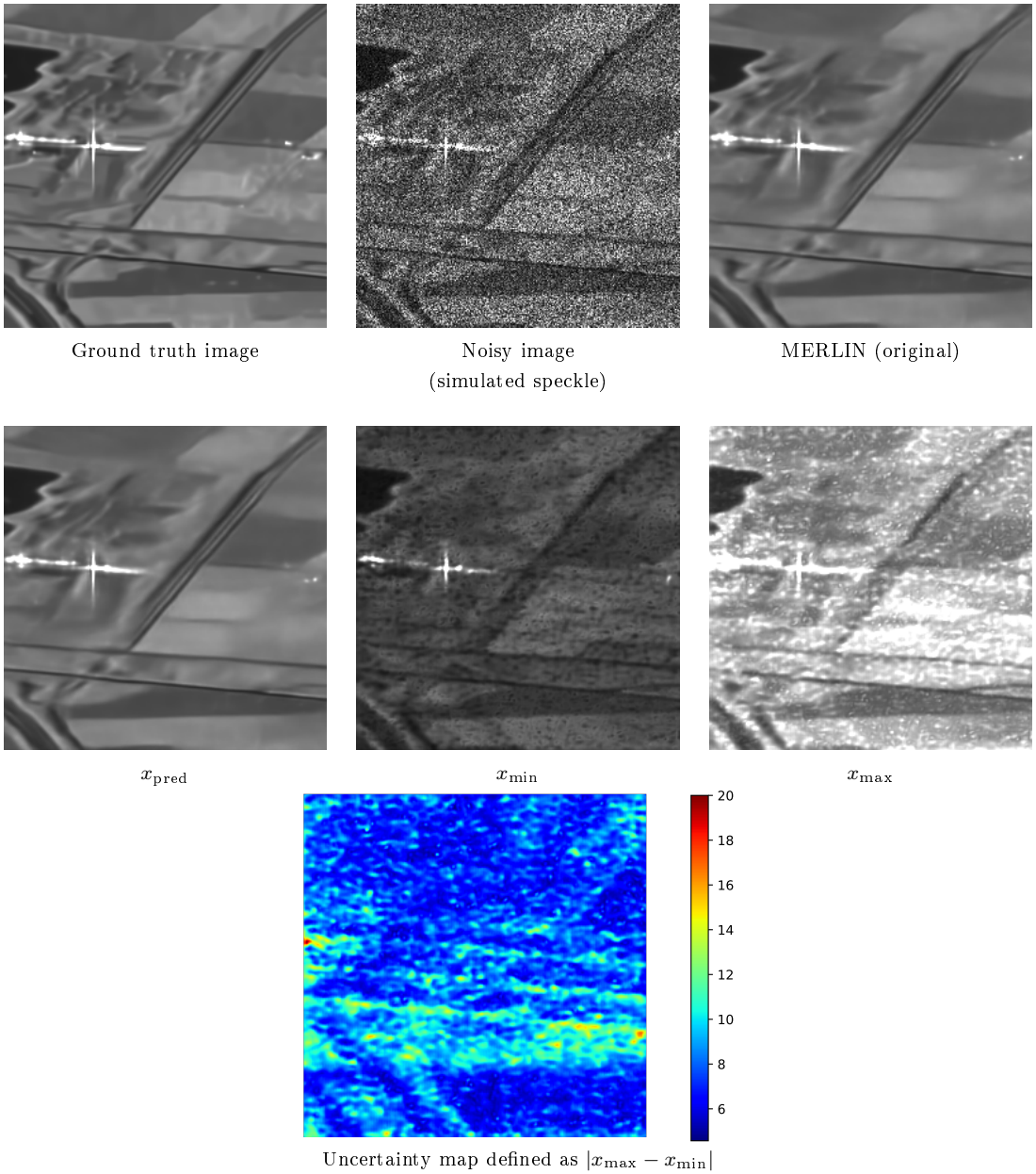


Figure 6.5: Predictions of the network when trained to optimize the loss function corresponding to equation 6.9 when x_{pred} has been pre-estimated using the MERLIN network.

Inverse gamma distribution over the reflectivity for uncertainty quantification

The results obtained with a uniform distribution over the reflectivity were not satisfying. Based on existing work such as Speckle2Void [63], we tried to predict the parameter of an inverse gamma

distribution over the reflectivity[1]. The inverse gamma probability density function f at a pixel k is defined as

$$\begin{aligned} \mathrm{p}(r_k) &= f(r_k, \alpha, \beta) \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{r_k}\right)^{\alpha+1} \exp\left(-\frac{\beta}{r_k}\right) \end{aligned}$$

where α is the *shape* parameter and β is the *scale* parameter. We would rather to express these parameters using the relectivities[2] r_k and the estimated number of looks L. The inverse gamma is conjugated to the gamma distribution which guarantees a closed form of the equation 6.3.
First, we define the estimated reflectivities as the expectation of the distribution:

$$\mathbb{E}[r_k] \equiv \hat{r}_k = \frac{\beta}{\alpha - 1}$$

And for the variance, we have the following equation:

$$\mathrm{Var}[r_k] = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)},$$

so if we define L such that $\mathrm{Var}[r_k] = \frac{\hat{r}_k^2}{L}$ (like for the parameter L of the gamma distribution), we obtain the following expression of α and β for the inverse Gamma distribution:

$$\begin{aligned} \beta &= \hat{r}_k(\alpha - 1) \\ \alpha &= L + 2 \end{aligned} \tag{6.10}$$

Combining the equations 6.3 and 6.10 leads us to the expression of the following loss function:

$$\mathcal{L}(\mathcal{G}(L+2, \hat{r}_k(\alpha-1)), \hat{r}_k, b, L) = -\log\left[\frac{(\hat{r}_k(1+L))^{2+L}\left(b^2 + \hat{r}_k + L\hat{r}_k\right)^{-\frac{5}{2}-L}\Gamma\left(\frac{5}{2}+L\right)}{\sqrt{\pi}\,\Gamma(2+L)}\right] \tag{6.11}$$

The expression of the loss function $\mathcal{L}(\mathcal{G}(L+2, \hat{r}_k(\alpha-1)), \hat{r}_k, b, L)$ is complex this is why we will suppose that the network already estimated the value of $\hat{r}_k$ and we only want to estimate the parameter L. We did not manage to obtain a closed-form expression of the Maximum Likelihood estimator $\hat{L}_{\mathrm{ML}}$ for this approach.
We are expecting L to be quite high, indicating a low uncertainty. In the past, our network has predicted an output within a range very close to $[0, 1]$. To make it easier for the network, we express the predicted parameter $\hat{L}$ based on the output $f_\theta(\boldsymbol{a})_k$ of the network at pixel k in different ways:

$$\hat{L} = f_\theta(\boldsymbol{a})_k \tag{6.12}$$

$$\hat{L} = \frac{1}{f_\theta(\boldsymbol{a})_k} \tag{6.13}$$

$$\hat{L} = \exp\left(f_\theta(\boldsymbol{a})_k\right) \tag{6.14}$$

[1] Here we are working with reflectivity and $\boldsymbol{b}$, not in log scale.
[2] To have easier notations, we note $r_k = r_k(\boldsymbol{a})$

Trainings have been done in the exact same conditions to compare the different strategies/formulations to estimate L.

Experimental results are presented in Figure 6.6. We can see that the different expressions of the parameter L depending on the output of the network lead to a very similar map of the estimated number of looks. The map seems to depend on the reflectivity value because a high value of L is observed for areas of high reflectivity values. We are expecting a relatively high value of L on homogeneous zones, but obtained $L \approx 3$ which is far too low for the denoised images.

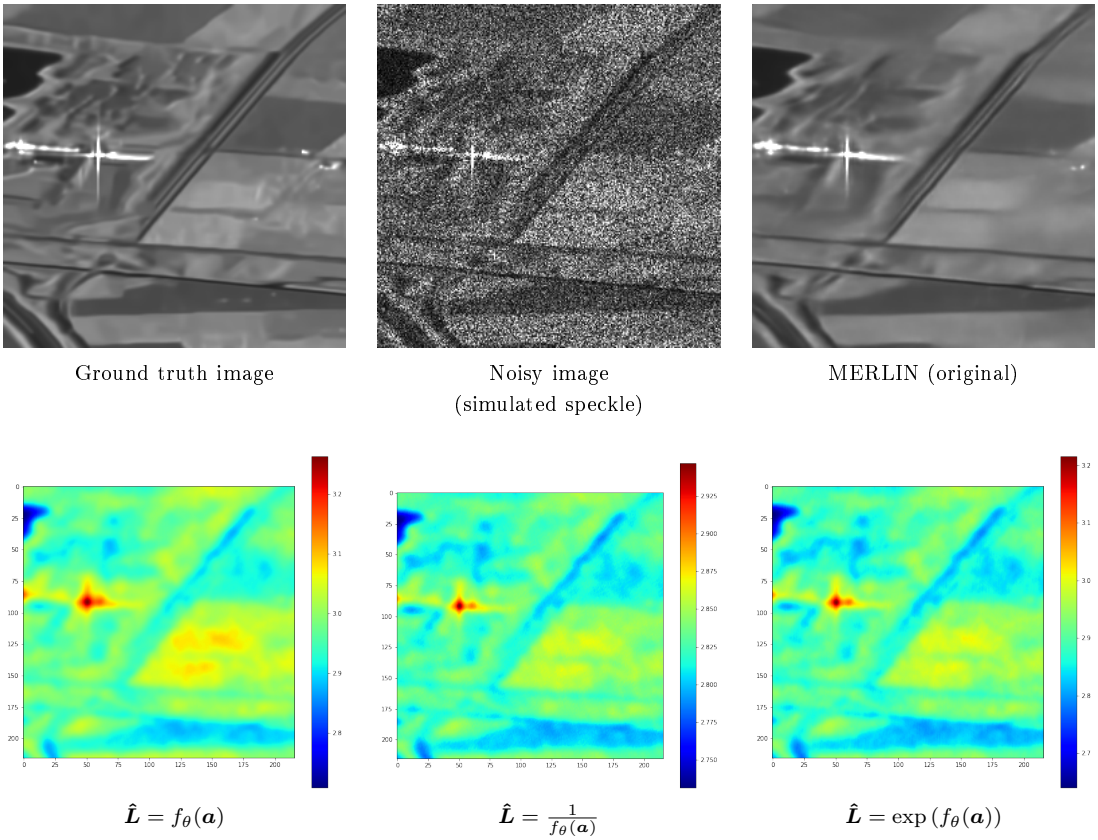


Figure 6.6: Experimental results with the inverse gamma distribution: the network estimates the parameter $\hat{L}$ for each pixel of the image. The larger $\hat{L}$, the smaller the variance and the more confident the network is on its prediction. Results using the three different parametrizations described in equations 6.14 are given.

Based on the previous results, we can see that assuming that the reflectivity follows a certain distribution (uniform or inverse gamma distributions) and training the network to estimate the parameters of this distribution does not work very well. The parameter maps are either noisy or too much correlated to the reflectivity values which is not a desired behavior. As the predicted parameters are directly linked to the variance of the reflectivity image, we can wonder whether the problem of estimating the variance of the reflectivity is not too hard to solve. This can be illustrated

by studying a lower bound of this estimator in an easier case: restoring images corrupted by an additive white Gaussian noise.

The variance estimation problem: illustration with additive Gaussian noise

We remind some definitions used in the following.

Let $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_d)^T \in \mathbb{R}^d$. We want to estimate $\boldsymbol{\theta}$ based on N measurements noted $i_1, i_2, \ldots, i_N$, and each measurement is independently distributed according to the probability density function $f(i; \boldsymbol{\theta})$. The Fisher information matrix $\mathbf{I} \in \mathbb{R}^{d \times d}$ is defined as

$$I_{m,n} = -\mathbb{E}\left[\frac{\partial^2}{\partial\theta_m \, \partial\theta_n} \log f(i; \boldsymbol{\theta})\right] \text{ for all } m, n \in 1, \ldots, d \tag{6.15}$$

Let $\hat{\boldsymbol{\theta}} = \left(\hat{\theta_1}, \hat{\theta_2}, \ldots, \hat{\theta_d}\right)^T \in \mathbb{R}^d$ be an **unbiased** estimator of $\boldsymbol{\theta}$. The Cramér-Rao bound states that the covariance of $\hat{\boldsymbol{\theta}}$ is always greater or equal to the inverse of the Fisher information matrix:

$$\mathrm{cov}_{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}) \geq \boldsymbol{I}(\boldsymbol{\theta})^{-1} \tag{6.16}$$

We also have, for all $m \in 1, \ldots, d$

$$\mathrm{var}(\hat{\theta_m}) \geq \left[\boldsymbol{I}(\boldsymbol{\theta})^{-1}\right]_{m\,m} \tag{6.17}$$

To make things easier, we work with Gaussian noise.
Let r be the reflectivity such that

$$r \sim \mathcal{N}(\mu, \sigma_r^2)$$

where σ_r defines the uncertainty on r.
We define the observed image as $i = r + \epsilon$ with

$$\epsilon \sim \mathcal{N}(0, \sigma_b^2)$$
$$\text{so that } i|r \sim \mathcal{N}(r, \sigma_b^2)$$

Proposition 4. *The minimum relative error on the estimated parameter σ_r^2 is given by*

$$\begin{aligned} Err[\hat{\sigma_r^2}] &= \frac{\sqrt{Var[\hat{\sigma_r^2}]}}{\sigma_r^2} \\ &= \sqrt{\frac{2}{N}} \, \frac{\sigma_r^2 + \sigma_b^2}{\sigma_r^2} \end{aligned}$$

In our case, we have $\sigma_r^2 \ll \sigma_b^2$, and the relative error can thus be written

$$Err[\hat{\sigma_r^2}] \approx \sqrt{\frac{2}{N}} \, \frac{\sigma_b^2}{\sigma_r^2}$$

When the number of samples N increases, the relative error decreases. However, the term $\frac{\sigma_b^2}{\sigma_r^2}$ is really high. This illustrates that estimating the small variance of a random variable when only accessing to observations that are the sum of this small-variance random variable plus another random variable with much larger variance is difficult and requires a very large amount of samples (if $\sigma_b = 100\,\sigma_r$, N has to be 10^8 times larger than when $\sigma_r^2 = \sigma_b^2$) This may explain why the problem we are trying to solve is difficult and the network does not manage to estimate reliably Δx or the variance parameter L when we are using a uniform or an inverse gamma distribution for the reflectivity. Training with huge data sets and much larger mini-batch sizes may significantly improve the performance. We followed a different direction instead that works on small training sets.

Proof. The unconditional distribution followed by i is also Gaussian:

$$i \sim \mathcal{N}(\mu, \sigma_r^2 + \sigma_b^2)$$

The uncertainties we want to estimate are related to r. We suppose that we know σ_b^2 (a characteristic of the observation system). We want to estimate μ and σ_r^2.
The Probability Density Function f of i is defined as

$$f(i; \mu, \sigma_r^2) = \frac{1}{\sqrt{2\,\pi}} \; \frac{1}{\sqrt{\sigma_r^2 + \sigma_b^2}} \; \exp\left(-\frac{1}{2} \; \frac{(i-\mu)^2}{\sigma_r^2 + \sigma_b^2}\right) \tag{6.18}$$

The negative log-likelihood $\mathcal{L}$ is defined as

$$\begin{aligned} \mathcal{L}(i; \mu, \sigma_r) &= -\log f(i; \mu, \sigma_r^2) \\ &= \frac{1}{2} \times \log(\sigma_r^2 + \sigma_b^2) + \frac{1}{2} \; \frac{(i-\mu)^2}{\sigma_r^2 + \sigma_b^2} + \log(\sqrt{2\,\pi}) \end{aligned} \tag{6.19}$$

We consider N independent and identically distributed samples $i_1, i_2, \ldots, i_N$. The joint log-likelihood is:

$$\begin{aligned} -\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2) &= \sum_{k=1}^{N} -\log f(i_k; \mu, \sigma_r^2) \\ &= \sum_{k=1}^{N} \left[\frac{1}{2} \times \log(\sigma_r^2 + \sigma_b^2) + \frac{1}{2} \; \frac{(i_k-\mu)^2}{\sigma_r^2 + \sigma_b^2} + \log(\sqrt{2\,\pi})\right] \\ &= N \; \log(\sqrt{2\,\pi}) + \sum_{k=1}^{N} \left[\frac{1}{2} \times \log(\sigma_r^2 + \sigma_b^2) + \frac{1}{2} \; \frac{(i_k-\mu)^2}{\sigma_r^2 + \sigma_b^2}\right] \end{aligned} \tag{6.20}$$

and corresponds to the empirical expectation of the single-observation neg-log likelihood up to a constant.
We define $\hat{\mu}^{(\mathrm{ML})}$ by

$$\hat{\mu}^{(\mathrm{ML})} = \arg\min_{\mu} -\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)$$

The first order derivative of the negative log-likelihood is given by

$$\frac{\partial}{\partial \mu}\left(-\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)\right) = \frac{1}{\sigma_r^2 + \sigma_b^2} \sum_{k=1}^{N} (\mu - i_k)$$

The estimator $\hat{\mu}^{(\mathrm{ML})}$ is computed by solving $\frac{\partial}{\partial \mu}\left(-\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)\right) = 0$:
leading to

$$\hat{\mu}^{(\mathrm{ML})} = \frac{1}{N} \sum_{k=1}^{N} i_k \tag{6.21}$$

We find the expression of the empirical mean, as expected.

In the exact same way, we define $\hat{\sigma_r^2}^{(\mathrm{ML})}$ by

$$\hat{\sigma_r^2} = \min_{\sigma_r^2}\left(-\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)\right)$$

Note that σ_r^2 is treated as a variable, not as the square of a variable in the following derivations. The notation σ_r^2 is kept to prevent the introduction of additional notations.

$$\frac{\partial}{\partial \sigma_r^2}\left(-\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)\right) = \frac{1}{2} \frac{1}{\sigma_r^2 + \sigma_b^2} \left[N - \sum_{k=1}^{N} \frac{(i_k - \mu)^2}{\sigma_r^2 + \sigma_b^2}\right]$$

$\hat{\sigma_r^2}^{(\mathrm{ML})}$ verifies $\frac{\partial}{\partial \sigma_r^2}\left(-\log f(i_1, i_2, \ldots, i_N; \mu, \sigma_r^2)\right) = 0$:
Leading to

$$\hat{\sigma_r^2}^{(\mathrm{ML})} = \frac{1}{N} \sum_{k=1}^{N} (i_k - \mu)^2 - \sigma_b^2$$

Again, we find the empirical variance corrected by the noise variance, a result that was expected.

We now compute the Fisher information matrix. In our case, the parameters vector $\boldsymbol{\theta}$ is defined as $\boldsymbol{\theta} = (\mu, \sigma_r^2)$.
The Fisher information matrix is defined as

$$\mathbf{I}(\mu, \sigma_r^2) = \begin{pmatrix} \mathbf{E}\left[\frac{\partial^2}{\partial \mu^2} \mathcal{L}(i; \mu, \sigma_r^2)\right] & \mathbf{E}\left[\frac{\partial^2}{\partial \mu \, \partial \sigma_r^2} \mathcal{L}(i; \mu, \sigma_r^2)\right] \\ \mathbf{E}\left[\frac{\partial^2}{\partial \sigma_r^2 \, \partial \mu} \mathcal{L}(i; \mu, \sigma_r^2)\right] & \mathbf{E}\left[\frac{\partial^2}{\partial {\sigma_r^2}^2} \mathcal{L}(i; \mu, \sigma_r^2)\right] \end{pmatrix}$$

Let us compute each term of the matrix:

$$\mathbf{E}\left[\frac{\partial^2}{\partial \mu^2} \mathcal{L}(i; \mu, \sigma_r^2)\right] = \frac{N}{\sigma_r^2 + \sigma_b^2}$$

$$\mathbf{E}\left[\frac{\partial^2}{\partial \sigma_r^2 \, \partial \mu} \mathcal{L}(i; \mu, \sigma_r^2)\right] = \mathbf{E}\left[\frac{i - \mu}{(\sigma_r^2 + \sigma_b^2)^2}\right] = 0$$

$$\mathbf{E}\left[\frac{\partial^2}{\partial\mu\ \partial\sigma_r^2}\ \mathcal{L}(i;\mu,\sigma_r^2)\right] = \mathbf{E}\left[\frac{i-\mu}{(\sigma_r^2+\sigma_b^2)^2}\right] = 0$$

$$\begin{aligned}\mathbf{E}\left[\frac{\partial^2}{\partial{\sigma_r^2}^2}\ \mathcal{L}(i;\mu,\sigma_r^2)\right] &= \frac{\mathbf{E}\left[(i-\mu)^2\right]}{(\sigma_r^2+\sigma_b^2)^3} - \frac{N}{2}\ \frac{1}{(\sigma_r^2+\sigma_b^2)^2} \\ &= \frac{N}{2}\ \frac{1}{(\sigma_r^2+\sigma_b^2)^2}\end{aligned}$$

Finally, we can write the Fisher information matrix as follows

$$\mathbf{I}(\mu,\sigma_r^2) = \begin{pmatrix} \frac{N}{\sigma_r^2+\sigma_b^2} & 0 \\ 0 & \frac{N}{2}\ \frac{1}{(\sigma_r^2+\sigma_b^2)^2} \end{pmatrix} \tag{6.22}$$

Based on the definition of the Cramér-Rao bound and (6.17), we can derive a lower bound of the variance of our unbiased estimator $\hat{\sigma_r^2}$:

$$\mathrm{Var}[\hat{\sigma_r^2}] \geq \frac{2}{N}(\sigma_r^2+\sigma_b^2)^2$$

Note that the maximum likelihood estimator is asymptotically efficient, so the bound is reached when N is large.

Finally, the minimum relative error on the estimated parameter σ_r^2 is given by

$$\begin{aligned}\mathrm{Err}[\hat{\sigma_r^2}] &= \frac{\sqrt{\mathrm{Var}[\hat{\sigma_r^2}]}}{\sigma_r^2} \\ &= \sqrt{\frac{2}{N}}\ \frac{\sigma_r^2+\sigma_b^2}{\sigma_r^2}\end{aligned}$$

□

6.2.2 Prediction of the expected difference between the log reflectivities

This section presents an empirical method to estimate an uncertainty map based on the predictions of the MERLIN network [19].

Proposed approach

In the test phase, the MERLIN network provides two estimations of the reflectivities: the first one is only based on the real part $\boldsymbol{a}$ of the image and will be noted $\boldsymbol{r}_a$; the second one is only based on the imaginary part $\boldsymbol{b}$ and will be noted $\boldsymbol{r}_b$. The final estimation of the reflectivity image is the mean of these two estimations

$$\hat{\boldsymbol{r}}_{\text{final}} = \frac{\boldsymbol{r}_a + \boldsymbol{r}_b}{2}. \tag{6.23}$$

We can use these two estimations to provide an information on the uncertainty of the prediction: if $\boldsymbol{r}_a$ and $\boldsymbol{r}_b$ are really different, then the variability of the network prediction is high and so is the associated uncertainty. We can study the quantity

$$\frac{1}{2}\ (\log \boldsymbol{r}_a - \log \boldsymbol{r}_b)^2 .$$

One can argue that we could simply compute the difference between the two estimated reflectivity images every time the MERLIN network gives a prediction. Examples of such maps are provided in Figure 6.7 with simulated speckle. We can see that the difference maps are very noisy and it is difficult to have a good estimation of what we can call the *estimated-reflectivity difference*.

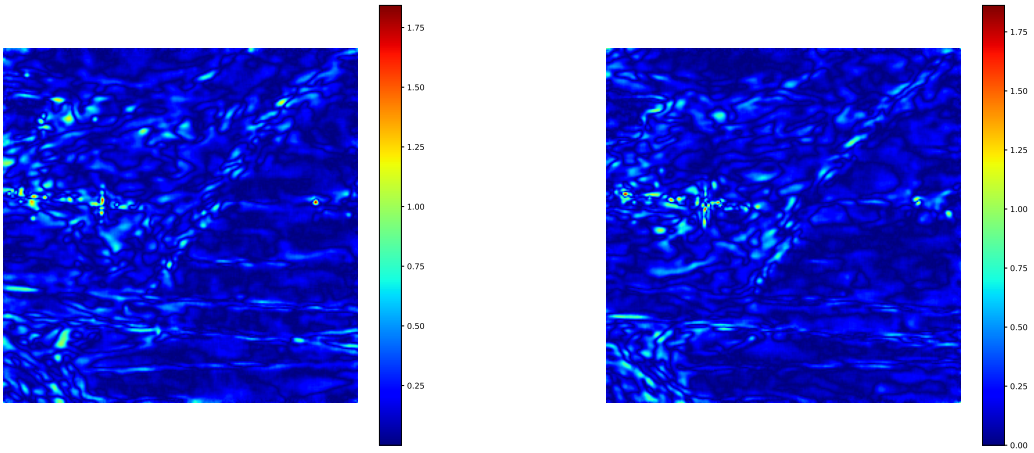


Figure 6.7: Raw difference maps defined as $\frac{1}{2}\ (\log \boldsymbol{r}_a - \log \boldsymbol{r}_b)^2$, computed based on 2 simulated SLC images of the same ground-truth reflectivity image.

We are then interested in predicting the expectation of the estimated-reflectivities difference noted d and defined as

$$d = \mathbb{E}\left[\frac{1}{2}\ (\log \boldsymbol{r}_a - \log \boldsymbol{r}_b)^2\right] \tag{6.24}$$

An approximation of this expectation has been computed throughout the following steps: first, we simulate speckle $N = 1000$ times over the same ground truth image. Then, for each noisy sample k, we estimate the images $\boldsymbol{r}_{a,\,k}$ and $\boldsymbol{r}_{b,\,k}$ with the original MERLIN network trained on simulated speckle. For each noisy sample k, we compute the estimated-reflectivities difference as $\frac{1}{2}\ (\log \boldsymbol{r}_{a,\,k} - \log \boldsymbol{r}_{b,\,k})^2$. We finally average all the estimated-reflectivities difference: $\sum_{k=1}^{N} \frac{1}{2}\ (\log \boldsymbol{r}_{a,\,k} - \log \boldsymbol{r}_{b,\,k})^2$. This map is shown figure 6.8.

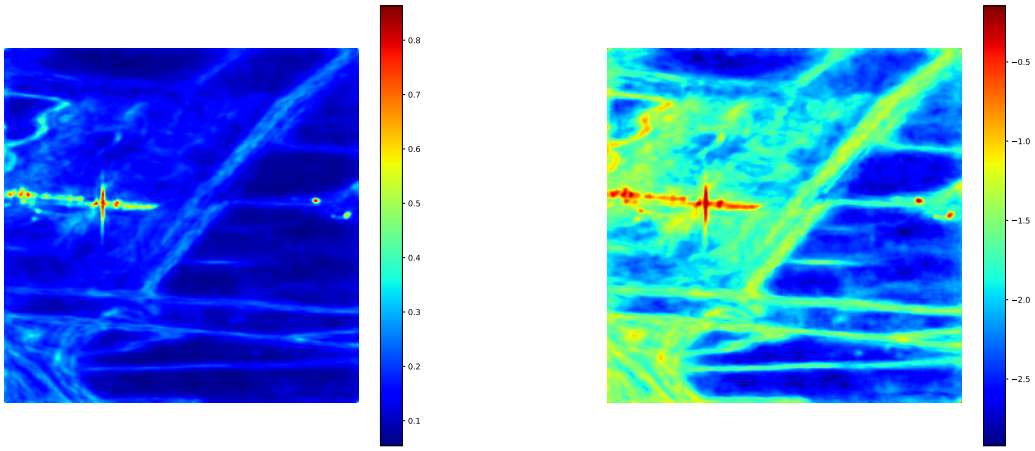


Figure 6.8: Average of all the estimated-reflectivities distance: $\sum_{k=1}^{N} \frac{1}{2} (\log \boldsymbol{r}_{a,\,k} - \log \boldsymbol{r}_{b,\,k})^2$; left: linear scale; right: log scale.

We suggest to train a network to predict d (see equation 6.24).
The network takes as input the real part $\boldsymbol{a}$ and the reflectivity predicted by the MERLIN network based on the real part $\boldsymbol{r}_a$ (equivalently $\boldsymbol{b}$ and $\boldsymbol{r}_b$), and predicts the expected difference between the log-reflectivity images in equation 6.24. The estimation is noted $\hat{d}_a$ (and equivalently $\hat{d}_b$).
The Loss is then defined as follows

$$\begin{aligned} \mathcal{L}(\boldsymbol{a}, \boldsymbol{r}_a, \boldsymbol{r}_b) &= \left|\left| \frac{1}{2} (\log \boldsymbol{r}_a - \log \boldsymbol{r}_b)^2 - \hat{d}_a \right|\right|^2 \\ &= \left|\left| \frac{1}{2} (\log \boldsymbol{r}_a - \log \boldsymbol{r}_b)^2 - f_\theta(\boldsymbol{a}, \boldsymbol{r}_a) \right|\right|^2 \end{aligned} \tag{6.25}$$

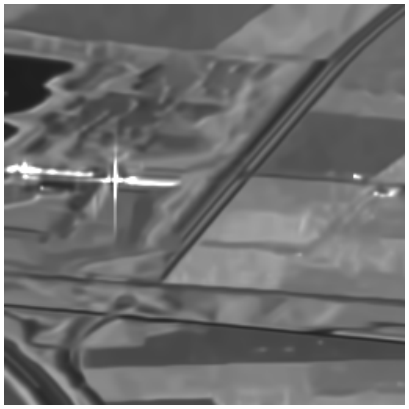
where $f_\theta(\boldsymbol{a}, \boldsymbol{r}_a)$ is the output of the network f_θ.

In test phase, the final estimation of the difference noted $\hat{d}$ is defined as

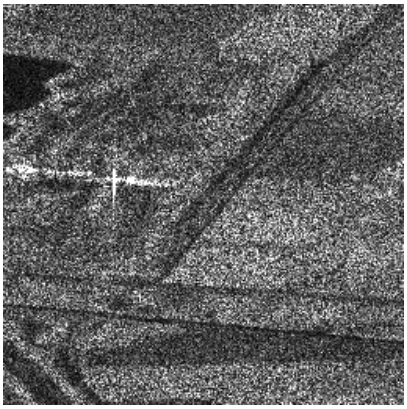
$$\hat{d} = \frac{\hat{d}_a + \hat{d}_b}{2} \tag{6.26}$$

Experimental results

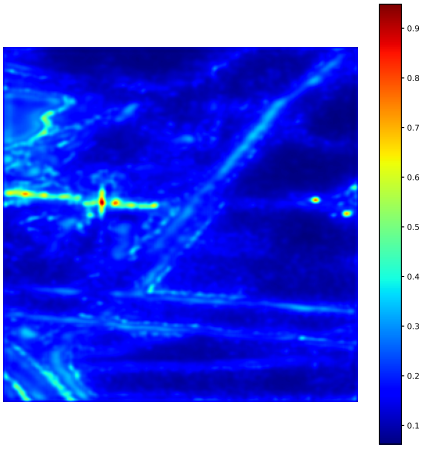
This method has proven to be effective and provide a good estimation of theoretical uncertainty map presented in Figure 6.8. The results shown in Figure 6.9 highlight the lack of confidence of the network concerning the edges and the bright scaterrers.



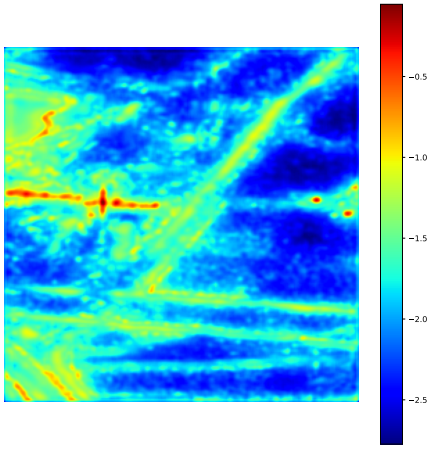
Groundtruth image



Noisy image (simulated speckle)



Estimated uncertainty map
in linear scale



Estimated uncertainty map
in log scale

Figure 6.9: Experimental results of the training of the network to minimize the loss function introduced in equation 6.25

Variance estimation

We want to justify why the formulation of the estimated-reflectivities distance leads to better results. This is why we derive in the following the relative error of the variance of our new estimator.

Once again, we consider the easier case of an additive Gaussian noise. Let $\boldsymbol{r}_a$ be the reflectivity predicted based on the real part and $\boldsymbol{r}_b$ the one predicted based on the imaginary part. We have

$$\begin{aligned}\boldsymbol{r}_a &\sim \mathcal{N}(\mu, \sigma_r^2)\\ \boldsymbol{r}_b &\sim \mathcal{N}(\mu, \sigma_r^2)\end{aligned}$$

Proposition 5. *The relative error of the variance of the estimator* $\frac{1}{2}(\boldsymbol{r}_a - \boldsymbol{r}_b)^2$ *is defined as*

$$\frac{\sqrt{Var\left[\frac{1}{2}\left(\boldsymbol{r}_a - \boldsymbol{r}_b\right)^2\right]}}{\sigma_r^2} = 2\sqrt{5} \tag{6.27}$$

Similarly, the relative error of the empirical variance $\bar{\sigma}^2$ *of the estimator* $\frac{1}{2}(\boldsymbol{r}_a - \boldsymbol{r}_b)^2$ *computed based on* N *independent samples* $\Delta_1, \Delta_2, \ldots, \Delta_N$ *is defined as*

$$\frac{\sqrt{Var\left[\bar{\sigma}^2\right]}}{\sigma_r^2} = \sqrt{\frac{2}{N}} \tag{6.28}$$

The relative error in equation 6.27 is a constant and does not depend on the value of σ_r, meaning that these data are more reliable to perform the training of our network.

The relative error in equation 6.28 is not constant but is bounded. We find again that the error decreases when the number of samples N increases. But in this case, the number of samples N is not the same as the one used in the first part with the negative log-likelihood minimization: in the first case, we consider the real part and the imaginary part as two different samples; in the second case, we need them both to compute the estimated-reflectivity distance. If we note N_{ML} the number of samples with the first approach, we have $N = N_{\mathrm{ML}}/2$.

The relative error on σ_r^2 is much lower in the regime where $\sigma_b^2 \gg \sigma_r^2$ (our case, since the equivalent number of looks after denoising is much larger than the initial number of looks). This means that much fewer samples are required to train the network (for each epoch and each mini-batch).

Proof. We want to evaluate the signal-to-noise-ratio of the raw differences between two estimated reflectivities:

$$\frac{\sqrt{\mathrm{Var}\left[\frac{1}{2}\left(\boldsymbol{r}_a - \boldsymbol{r}_b\right)^2\right]}}{\sigma_r^2}$$

First, we compute the expectation of the estimated-reflectivity distance

$$\mathbb{E}\left[\left(\boldsymbol{r}_a - \boldsymbol{r}_b\right)^2\right] = \mathbb{E}\left[\boldsymbol{r}_a^2\right] + \mathbb{E}\left[\boldsymbol{r}_b^2\right] - 2\mathbb{E}\left[\boldsymbol{r}_a \;\; \boldsymbol{r}_b\right]$$

$\boldsymbol{r}_a$ and $\boldsymbol{r}_b$ are iid so $\mathbb{E}\left[\boldsymbol{r}_a \;\; \boldsymbol{r}_b\right] = \mu^2$, leading to

$$\begin{aligned}\mathbb{E}\left[(\boldsymbol{r}_a - \boldsymbol{r}_b)^2\right] &= \mathbb{E}\left[\boldsymbol{r}_a^2\right] + \mathbb{E}\left[\boldsymbol{r}_b^2\right] - 2\mu^2 \\ &= \sigma_r^2 + \mu^2 + \sigma_r^2 + \mu^2 - 2\mu^2 \\ &= 2\ \sigma_r^2\end{aligned}$$

By estimating the expectation of the estimated-reflectivity distance, we are estimating the variance of the reflectivity σ_r^2.
We want to compute the variance of our estimator. To make it easier to compute, we introduce $\Delta = \boldsymbol{r}_a - \boldsymbol{r}_b$, and

$$\Delta \sim \mathcal{N}(0, 2\ \sigma_r^2)$$

As Δ is a Gaussian random variable, we will use the known expression of the moments (of order 4 especially) in the following.

$$\begin{aligned}\mathrm{Var}\left[(\boldsymbol{r}_a - \boldsymbol{r}_b)^2\right] &= \mathrm{Var}\left[\Delta^2\right] \\ &= \mathbb{E}\left[\Delta^4\right] - \mathbb{E}\left[\Delta^2\right]^2 \\ &= 3(\sqrt{2}\,\sigma_r)^4 - (2\,\sigma_r^2)^2 \\ &= 20\,\sigma_r^4\end{aligned}$$

We finally compute the relative error:

$$\begin{aligned}\frac{\sqrt{\mathrm{Var}\left[\frac{1}{2}(\boldsymbol{r}_a - \boldsymbol{r}_b)^2\right]}}{\sigma_r^2} &= \frac{1}{2}\frac{\sqrt{\mathrm{Var}\left[(\boldsymbol{r}_a - \boldsymbol{r}_b)^2\right]}}{\sigma_r^2} \\ &= \frac{\sqrt{20}\,\sigma_r^4}{\sigma_r^2} \\ &= 2\sqrt{5}\end{aligned}$$

In order to keep the same framework as section 6.2.1, we also compute the relative error using the empirical variance computed based on N independent samples $\Delta_1, \Delta_2, \ldots, \Delta_N$
The empirical mean approximating the expression $\mathbb{E}\left[\Delta^2\right]$ is given by

$$\bar{\sigma}^2 = \frac{1}{2N}\sum_{k=1}^{N}\Delta_k^2$$

As Δ_k for $k \in \{1, \ldots, N\}$, is centered, we have

$$\begin{aligned}\mathbb{E}\left[\bar{\sigma}^2\right] &= \mathbb{E}\left[\frac{1}{2\,N}\sum_{k=1}^{N}\Delta_k^2\right] \\ &= \frac{1}{2}\,\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N}\Delta_k^2\right] \\ &= \sigma_r^2\end{aligned}$$

This is thus an unbiased estimator of σ_r^2 (like the ML estimator that worked on noisy samples i).

We want to compute

$$\frac{\sqrt{\text{Var}\left[\bar{\sigma}^2\right]}}{\sigma_r^2}$$

where $\bar{\sigma}^2$ is defined as a sum of N independent square Gaussian variables. We can re-write $\bar{\sigma}^2$ as

$$\begin{aligned}\bar{\sigma}^2 &= \frac{1}{2N} \sum_{k=1}^{N} \Delta_k^2 \\ &= \frac{1}{2\,N} \times 2\,\sigma_r^2 \sum_{k=1}^{N} \left(\frac{\Delta_k}{\sqrt{2}\,\sigma_r}\right)^2\end{aligned}$$

Since $\Delta_k \sim \mathcal{N}(0, 2\ \sigma_r^2)$ for all $k \in \{1, \ldots, N\}$, we deduce that

$$\frac{\Delta_k}{\sqrt{2}\,\sigma_r} \sim \mathcal{N}(0,1) \text{ for all } k \in \{1, \ldots, N\}$$

$\bar{\sigma}^2$ is then proportional to the sum of N independent square standard normal variables, meaning that it is distributed according to a Chi square distribution, with N degrees of freedom, and the variance follows from the closed-form expression of the variance of a χ^2 random variable:

$$\text{Var}\left[\sum_{k=1}^{N} \left(\frac{\Delta_k}{\sqrt{2}\,\sigma_r}\right)^2\right] = 2\,N$$

We can then compute the relative error

$$\begin{aligned}\frac{\sqrt{\text{Var}\left[\bar{\sigma}^2\right]}}{\sigma_r^2} &= \frac{\sqrt{\text{Var}\left[\frac{1}{2N}\ 2\,\sigma_r^2 \sum_{k=1}^{N} \left(\frac{\Delta_k}{\sqrt{2}\,\sigma_r}\right)^2\right]}}{\sigma_r^2} \\ &= \frac{1}{2N}\ 2\sigma_r^2\ \frac{\sqrt{\text{Var}\left[\sum_{k=1}^{N} \left(\frac{\Delta_k}{\sqrt{2}\,\sigma_r}\right)^2\right]}}{\sigma_r^2} \\ &= \frac{1}{2N}\ 2\sigma_r^2\ \frac{\sqrt{2\,N}}{\sigma_r^2} \\ &= \sqrt{\frac{2}{N}}\end{aligned}$$

□

6.3 Conclusion

In this chapter, we presented three methods to estimate uncertainties related to despeckling. The first two methods rely on the prediction of a distribution over the reflectivity for each pixel. Parameters of a uniform distribution and an inverse gamma distribution have been predicted by the network. The experimental results were not very satisfying, this could be explained by the computation of the relative error of the variance of our estimators: the relative error is very high which means that the problem we want to solve is difficult. A third method, simple yet effective, is proposed to compute an uncertainty map associated to the reflectivity images estimated with the MERLIN network. The methodological framework introduced in section 6.2.1 is very interesting but has led to poor experimental results. This can be explained by the high relative error associated with the variance estimation at the core of our approach.

Part III

Conclusion

Chapter 7

Conclusion and perspectives

7.1 Conclusion

This thesis presents our work on joint despeckling of SAR images and provides some methods to estimate the uncertainty related to their restoration.

Joint despeckling can be performed using two different strategies: the Multi-Input Multi-Output framework and the Multi-Input Single-Output framework. In chapter 3, we explained that the MIMO strategy can be used when the number of input images is low (equal to 2 or 3) whereas the MISO strategy can be used with as many input images as possible, the main limitation being the capacity of the network we plan to use. The MIMO has the advantage that only one forward pass is needed to estimate the reflectivity images associated to all the input images when the MISO strategy requires one forward pass for each image.

In this work, the MIMO strategy has been used for joint despeckling of dual polarization Sentinel-1 GRDM EW images for sea ice analysis. We proposed a self-supervised deep learning approach based on a previous method developed by the IMAGES team called SAR2SAR [18]. One of the main challenges related to sea ice images is the impact of thermal noise. We decided to feed our network with corrected images where the thermal noise floor bias has been removed with the Korosov algorithm [69]. The joint processing of HH and HV images improves the restoration of thin structures such as tiny rivers, and the processing of corrected images leads to less fluctuations in low reflectivity areas in the restored images. The artifact (vertical line) appearing when subtracting the thermal noise floor after the despeckling step is strongly reduced in our approach.

When considering applications on land (urban areas or agriculture monitoring), the thermal noise is negligible. Chapter 5 focuses on multi-temporal despeckling where several images of the same area acquired at different times are used. A simple way of processing the multi-temporal stack is to compute a temporal average image or even a super-image. Deep learning methods can be used within existing frameworks such as the Quegan filter [70] or the RABASAR framework [16]. An unsupervised deep learning method based on the MERLIN approach [19] is proposed in this chapter to reach high-quality despeckling even when few images are available. This method takes as input

the real or imaginary part of the reference image and intensity images of other dates in additional channels. Adding more images provides more temporal information and improves the quality of the despeckled images. This gain increases with respect to the number of input images, but its growth reaches a plateau. While this approach improves the quality of the estimated reflectivity images, its main limit is that a specific network needs to be trained for each number of input images.

The final contributions described in chapter 6 are related to uncertainty estimation for despeckling. Based on the MERLIN network, an uncertainty map is estimated to provide information on the network's confidence. The first approaches are based on the predictions of the parameters of a distribution over the reflectivity pixels. The networks provide a 1D distribution of reflectivities at each pixel. The sharper the distribution, the more confident the network is. The choice of such distribution is constrained by the need of a closed form of the loss function. Uniform and inverse gamma distributions were studied, both leading to poor results in practice. This is explained by a simplified study of the relative error of the variance of our estimator computed in the more simple case of additive white Gaussian noise.
Another method has been proposed and is based on the computation of the expected difference between the restored image based on the real part of the image, and the restored image based on the imaginary part. This approach leads to good results and the estimated uncertainty map shows high uncertainty on edges and areas containing bright scaterrers.

The architecture and the contributions of this thesis are summarized in Figure 7.1.

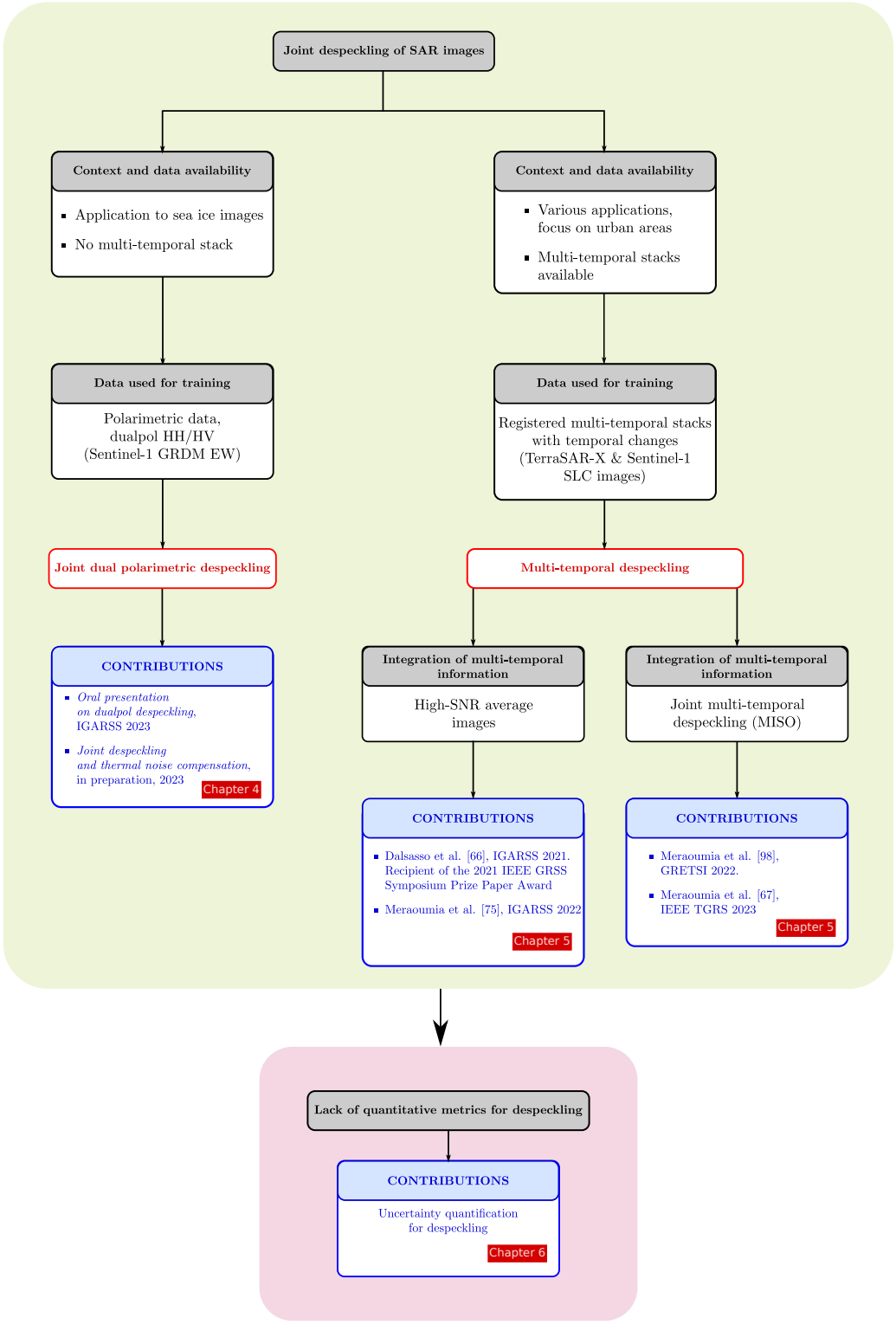


Figure 7.1: Overview of the scientific contributions of this thesis

7.2 Remaining issues and perspectives

The perspectives are divided into two categories in the following sections: methodological perspectives and application perspectives.

7.2.1 Methodological perspectives

Number of input images for multi-temporal despeckling

The multi-temporal unsupervised method developed in section 5.3 is not flexible in the sense that adaptation to a different number of input images requires the training of the method. It would be interesting to have a new framework where the testing phase can be conducted with an arbitrary number of input images corresponding to the images available for a specific application.

In preliminary experiments, wanted to perform an **early fusion** of the input images. We trained a network taking as input the noisy image and a super-image corresponding to the temporal mean computed with all the available dates. Thus, depending on how many images are available for the test phase, the level of the speckle of this multi-temporal average fluctuates, but is lower that the level of speckle of the image we want to restore. It also provides information on stable structures.

To test this hypothesis, we considered simulated speckle on ground truth images computed by averaging all the images in a Sentinel-1 temporal stack, and then despeckling this temporal mean with MuLoG [14]. We work with 5 temporal stacks. These ground truth images have also been used in the training of the network SAR2SAR in the original paper [18]. The input of the network is a tensor composed of two images: the first image (the one we want to restore) is corrupted by a speckle with an equivalent number of look $L = 1$ (single look SAR image); the second image is the associated multi-temporal average computed by averaging 10 noisy samples from the Sentinel-1 stack [1].

The network is trained using the hyper-parameters of the original SAR2SAR network [18] in the training phase A. During the training, a pair of noisy images is simulated using the same patch from the ground truth image, and the multi-temporal average is computed for each patch. The loss function is computed based on the output of the network (i.e. the estimated despeckled image) and the second noisy sample (i.e. the target).

The experimental results are given in Figure 7.2.

[1] the speckle is not simulated, the noisy images come from a stack of single look Sentinel-1 images

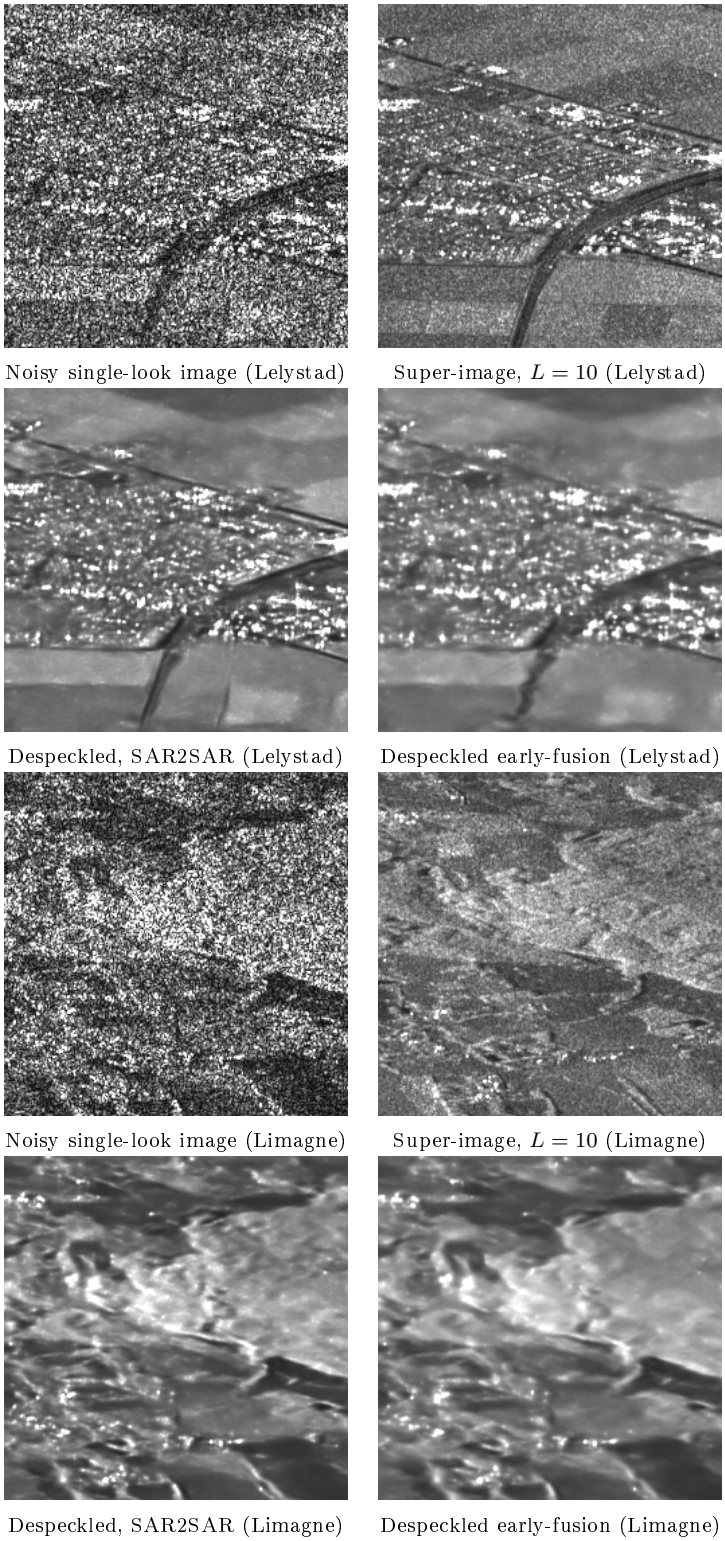


Figure 7.2: Results of the early fusion method en Sentinel-1 images of Lelystad, the Netherlands, and Limagne, France. The speckle is simulated for the noisy images, and the super-image is computed by averaging 10 Sentinel-1 images of the same area acquired at different times. The early fusion restored image is more blurred that the one restored with the original SAR2SAR network.

We can see that the early fusion does not give better results even though we provide the network with more information: the super-image has a low level of noise and the stable structures such as the field delimitations and the roads are more visible. These results were unexpected. This could be explained by the difficulty to process images with different level of noise when just concatenated before being fed to the network. A solution could be to provide an evaluation of the equivalent number of looks as in FFDNet [30].

Another strategy could be to perform an **iterative despeckling** as in FastDVDNet [93]. Five consecutive frames are used as input to denoise the central frame. Triplets of these consecutive frames are fed to a first denoising blocks, and all the blocks have the same weights. The outputs of theses blocks are then fed to a second denoising block, leading to the final estimation of the denoised middle frame. The proposed cascade architecture is shown in Figure 7.3.

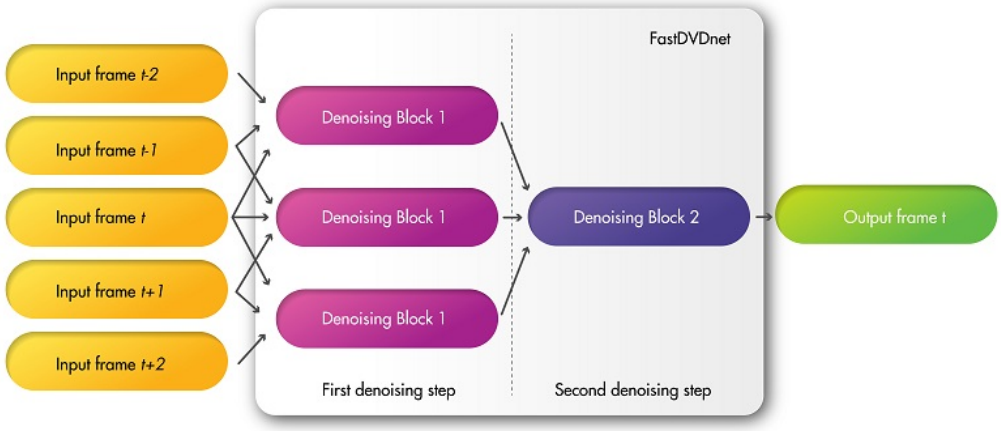


Figure 7.3: Cascade architecture of FastDVDnet. Five consecutive frames are taken as input of the framework. The first denoising blocks take three consecutive images as input, and share the weights. The second denoising step takes the three outputs of the previous blocks. The network estimated the denoised image corresponding to the central frame of the input sequence. Illustration from [93].

They also prove that this two-step denoising is improving the restored images: a network taking the five frames as input and predicting the denoised central frame is also trained. The quality of the estimations of the single denoising block network is worse that the one of the proposed approach, indicating that there may be room for improvement for our proposed multi-temporal despeckling approach.

Multi-modal and multi-sensor input images

In this thesis, we worked with images coming from a fixed sensor and the same modality. For some applications, it is useful to exploit as many images as possible even if they are not from the same sensors or the same modality.

Mixing images acquired with different sensors is not an easy task as the resolution can be very different. The incident angle is not the same either, meaning that the geometrical deformation are not the same. The network needs to be robust to these changes and exploit the shared information between images at different resolution. In the work of another PhD student of our research group [94], the joint exploitation of SAR images and optical images has been achieved using the MERLIN network. Even if the optical image is projected into the radar geometry, the deformation between the optical and radar images are different, meaning that this approach could generalize to other sensors.

Study of network architectures

As stated in chapter 3, the research of the best architecture for our problem has not been done in this thesis because this problem is far from trivial, time-consuming and should be tackled once the training strategy has proven itself efficient.

The effectiveness of the MIMO strategy is linked to the architecture: the more parameters we have and the deeper the network, the more input images the network will be able to handle.

Attention layers used in transformers could be used when performing multi-temporal despeckling. In the work on super resolution [95], a multi-temporal stack is taken as input and the network predicts a high resolution image and the related uncertainty map. The architecture is very complex and contains attention layers. A description of the architecture is shown in Figure 7.4.

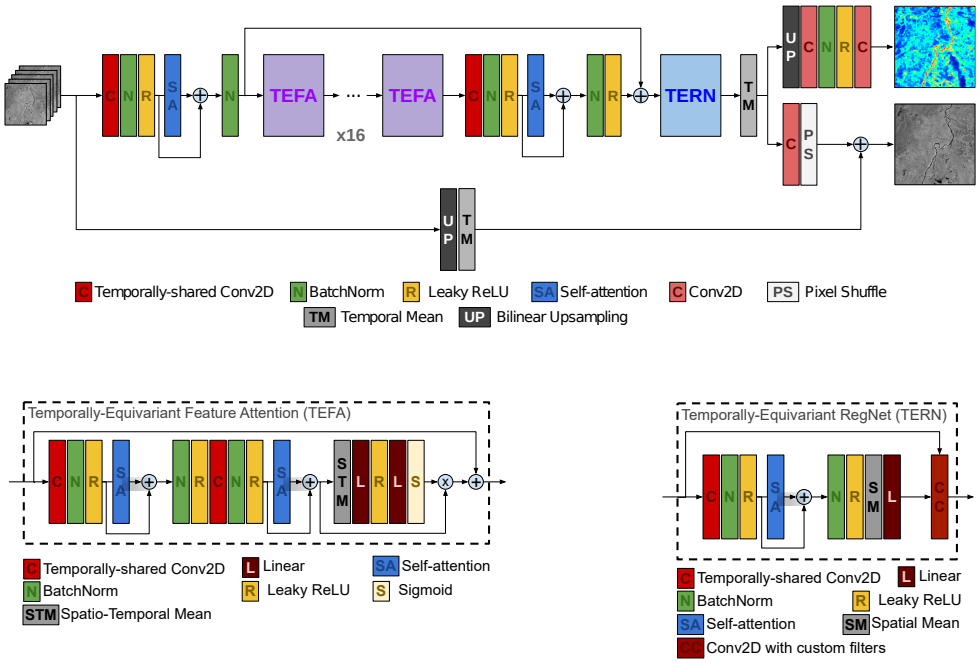


Figure 7.4: Architecture of the network PIUnet proposed in [95] for super-resolution. The model processes a stack of low resolution images and estimates an uncertainty map and a super-resolution image. The self-attention layers are key to enforce permutation invariance of the input images while being able to retrieve the stable structures within the temporal stack.

With attention layers, the authors enforce the permutation invariance of the input images. To build an invariant model, they use a sequence of equivariant operations followed by a global invariant function.

A function $f : X \to Y$ is said to be equivariant to the actions g of a group $\mathcal{G}$ if $f(g \circ x) = g \circ f(x)$ for all $x \in X, g \in \mathcal{G}$.

A function $f : X \to Y$ is said to be invariant to the actions g of a group $\mathcal{G}$ if $f(g \circ x) = f(x)$ for all $x \in X, g \in \mathcal{G}$.

In this case, they are dealing with the permutation group, and the actions are all the possible permutation of the temporal stack of images. For equivariant functions, the output will be the permuted version of the output corresponding to the order of the input images without the permutation. Having an invariant function at the end of the network means that the output will always be the same independently from the order of the input images. This means that the network is more robust and capture the input images properties, not their order. Building an entire invariant model is difficult because invariant functions are usually simple, the common example being the mean function. It is difficult to extract complex features only with invariant functions. The order of the images needs to be tracked to extract information and cross correlations between them, and then an invariant function can be applied. The authors of [95] have build a model composed of equivariant functions and ending with an invariant function. The equivariant function used in the network are self-attention.

In our work, it was approximately enforced by randomly shuffling the input channels during the training step. The capacity of attention layers to recover the invariant information within the stack of images make them more robust to permutations.

7.3 Application perspectives

7.3.1 Despeckling for sea ice classification

This work is done in collaboration with the UiT the Arctic University in Tromso and the researchers Debanshu Ratha, Johannes Lohse and Andrea Marinoni.

There is a large number of downstream applications of SAR remote sensing that may benefit from improved speckle reduction in S1 wide-swath imagery.

In this section, we focus on sea ice monitoring. Sea ice conditions are routinely mapped by national ice services worldwide and the resulting information is distributed in the form of ice charts. While operational ice chart production is at present still performed manually, multiple studies have investigated the (semi-)automated separation of sea ice and open water as well as the classification of different sea ice types, using both deep-learning approaches and statistical methods [96].

Here, we consider a pixel-wise classification algorithm introduced in [97].

The method uses the local incident angle together with both HH and HV backscatter intensities of Sentinel-1 GRDM EW images. It accounts for the well-known effect of class-dependent backscatter variation with local incident angle. A linearly variable mean value for each class distribution is assumed. We use a version of the classifier that was specifically trained for the area around Belgica Bank in Western Fram Strait, which is shown in Figure 7.5.

It distinguishes four ice classes (*Open Water*, *Young Ice*, *Level Ice*, *Deformed Ice*) which are relevant for tactical navigation and used in [98].

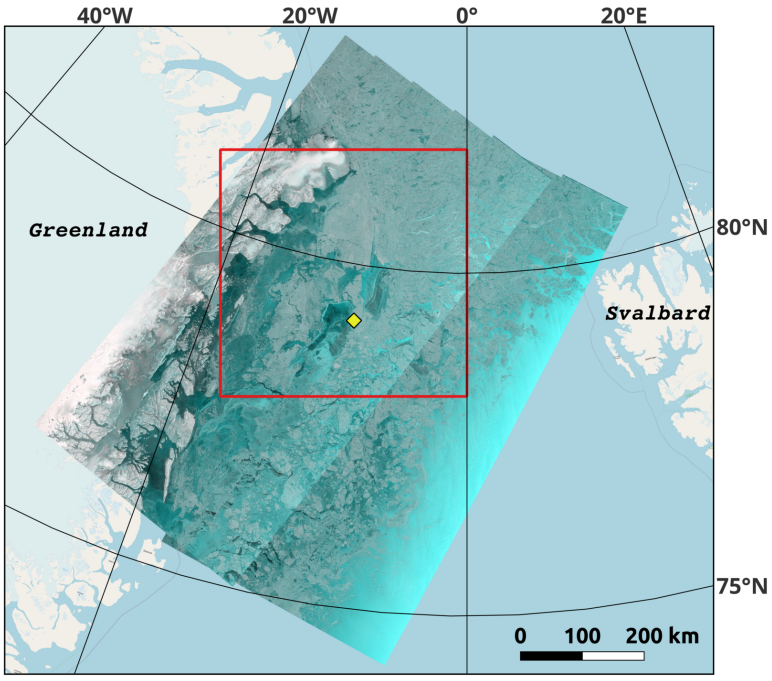


Figure 7.5: Area of interest for sea ice classification around Belgica Bank (red rectangle). The yellow marker indicates the position of the ship during the research expedition of the Norwegian Ice Service. The displayed image in RGB is computed using the HV, HH, HH. The images used in the study were acquired during two overpasses on 02/05/2022 and 03/05/2022.

We illustrate the positive impact of a despeckling step on the ice classification results. Because we do not have ground truth classifications, having a closer look at the performance of other methods can help validate our method.

For this experiment, we select four images from two Sentinel-1 overpasses on May 2nd and May 3rd 2022. We separately apply six different despeckling methods, including no speckle reduction, multi-looking with two different window sizes (9x9 and 21x21 pixels), the two baseline methods (MuloG with BM3D (d) and SAR-BM3D (e)), as well as the dual-polarimetry joint despeckling method described in section 4.2.3 (f).

The entire area and its corresponding sea ice classification is given in Figure 7.6. Zoomed-in close-up comparisons of the classification results obtained from the different speckle reduction methods are presented in Figure 7.7 and Figure 7.8.

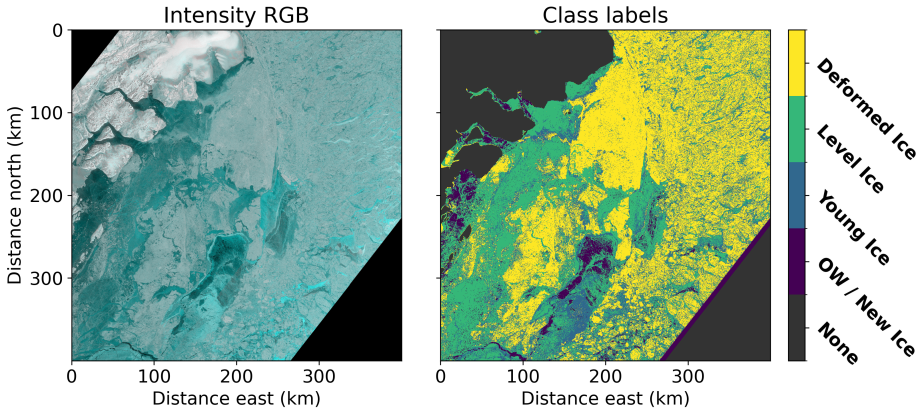


Figure 7.6: (left): speckle-free image of our area of interest estimated by the dual-polarimetry joint despeckling method network from section 4.2.3. The image is displayed using a false color RGB displaying (RGB: HV, HH, HH). (right): corresponding classification result after applying the dual-polarimetry joint despeckling method.

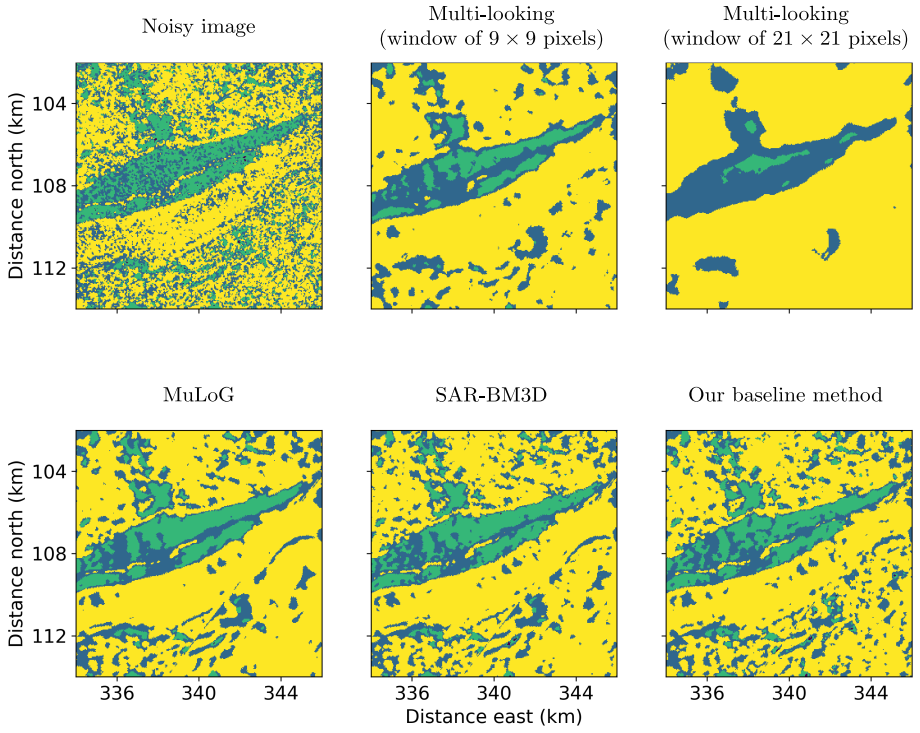


Figure 7.7: Close-up comparison of ice type classification results after different despeckling methods.

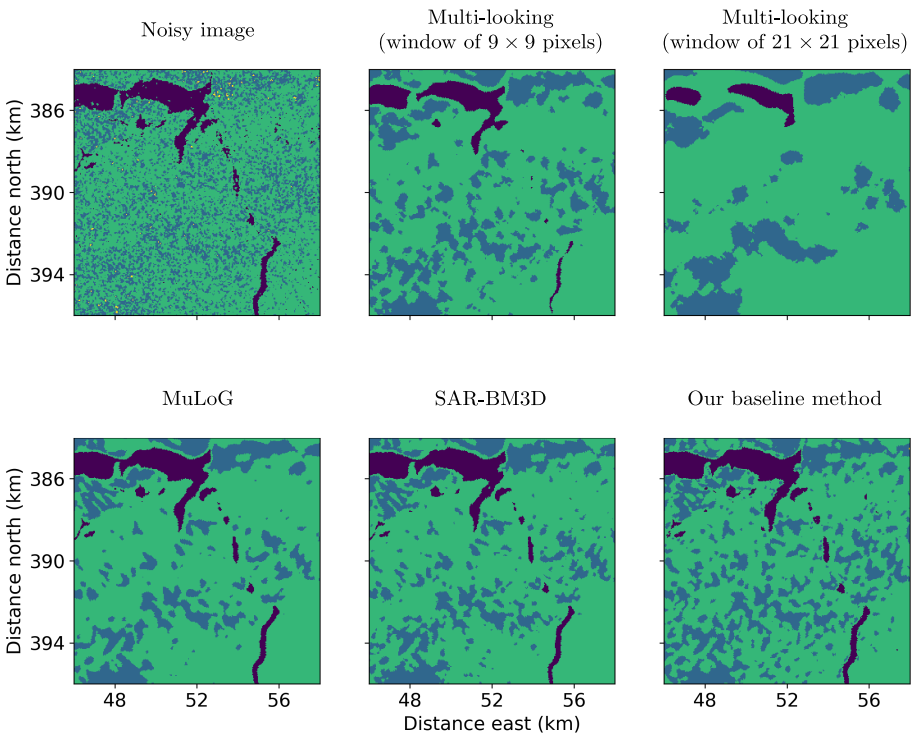


Figure 7.8: Close-up comparison of ice type classification result after different despeckling methods. The dual-polarimetry joint despeckling method enables the classifier to maintain small scale details such as the open lead in the lower right corner.

In Figure 7.7, the results clearly show that despeckling with large multi-looking windows blurs out class boundaries, in particular for relatively small spatial structures, resulting in large areas of wrong ice types. Our dual-polarimetry joint despeckling method is able to maintain the spatial detail, while at the same time reducing speckle effects significantly.

As expected, MuLoG, SARBM3D and our dual-polarimetry joint despeckling method outperform the multi-looking approaches. For our network, the classification results appear significantly smoother compared to no speckle reduction, while at the same time maintaining small-scale structures such as narrow leads which are important features for tactical navigation.

However, we can see the line artifact in Figure 7.7: the line is not vertical anymore because the image have been geocoded i.e. projected on the Earth and they are not in the radar geometry anymore. For geocoding, rotation are performed and have changed the orientation of the artifact line. This line comes from the correction step which removes the thermal noise component of the image. The correction is performed after despeckling. Our dual-polarimetry joint despeckling method has attenuated this artifact, but it is still there.

In a very forthcoming work, we want to train the classifier on the final network developed in section 4.3 which takes the thermal noise into account.

More generally, the despeckling step if often needed for applications using SAR images. Multi-

temporal despeckling is particularly useful when a large volume of data is available, and could thus be applied for any kind of Earth monitoring.

It could particularly be interesting to apply the methods developed in this thesis for forest monitoring, or tiny rivers detection as introduced in [75].

List of publications, seminars and awards

I Publications

- *Despeckling of dual-pol grd sentinel-1 images in extra-wide mode by deep learning* (Oral presentation), Ines Meraoumia, Debanshu Ratha, Emanuele Dalsasso, Loïc Denis, Florence Tupin, Andrea Marinoni, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference)**, 2023.
- *Joint despeckling and thermal noise compensation, application to Sentinel-1 images of the Arctic*, Ines Meraoumia, Debanshu Ratha, Emanuele Dalsasso, Johannes Lohse, Florence Tupin, Andrea Marinoni, Loïc Denis, **preprint**, 2023
- *Exploiting multi-temporal information for improved speckle reduction of Sentinel-1 SAR images by deep learning*, Emanuele Dalsasso, Ines Meraoumia, Loic Denis, Florence Tupin, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference), 2021**
  Recipient of the 2021 IEEE GRSS Symposium Prize Paper Award
- *Fast strategies for multi-temporal speckle reduction of Sentinel-1 GRD images*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Florence Tupin, **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference), 2022**
- *Débruitage multi-temporel d'images radar à synthèse d'ouverture par apprentissage profond auto-supervisé*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Florence Tupin, **GRETSI conference, 2022**
- *Multi-temporal speckle reduction with self-supervised deep neural networks*, Ines Meraoumia, Emanuele Dalsasso, Loic Denis, Remy Abergel, Florence Tupin, **IEEE Transactions on Geoscience and Remote Sensing, 2023**

II Seminars

- Multi-temporal despeckling of SAR images using the framework RABASAR and deep learning, Deep Learning Working Group seminar, LTCI Télécom Paris
- Multi-temporal despeckling of SAR images, Deep Learning Working Group seminar, LTCI Télécom Paris
- Multi-temporal despeckling of SAR images with self-supervised deep neural networks, Université Jean Monet, Saint-Etienne
- SAR image despeckling: an overview, UiT The Arctic University of Tromso, Norway
- Uncertainty quantification in deep learning, Deep Learning Working Group seminar, LTCI Télécom Paris
- Introduction to Remote Sensing: case study on despeckling Synthetic Aperture Radar images, Vertaix Lab, Princeton University, USA

III Awards

- **IEEE International Geoscience and Remote Sensing Symposium (IGARSS conference), 2022** for the paper *Exploiting multi-temporal information for improved speckle reduction of Sentinel-1 SAR images by deep learning*, Emanuele Dalsasso, Ines Meraoumia, Loic Denis, Florence Tupin

Appendix A

On normalizing the input of the network

A.1 Context of the work

In chapter 4, we are working with Sentinel-1 GRDM EW images (polarizations HH and HV). The main application of this work is the study of sea ice, but due to quick structural changes in these kind of areas, it is difficult to construct a temporal stack of images (needed to the training step of our network in order to select pairs of noisy images for our self-supervised method). The network is thus trained on land, but the will be tested on sea ice images. The reflectivity values of sea ice images is lower than the land images, this could lead to a distribution shift of the input patch. Neural networks are sensible to distribution shift, so we propose to evaluate the impact of a shift of reflectivity values on the quality of the restored images. The network used in the following is the SAR2SAR network taking one image as input (original framework), and we use the HH and HV images as 2 different samples.

A.2 Normalization applied in previous work

Based on [18] and even [19], the following normalization has been done before feeding the network with any image during the training and testing phases.
Let $\boldsymbol{A}$ be the amplitude image of interest. We want the network's input to be in $[0, 1]$ and we note $\boldsymbol{y}$ the normalized input image defined as:

$$\boldsymbol{y} = \frac{\log(\boldsymbol{A}) - m}{M - m}$$

where $m = \min \log(\boldsymbol{A})$ and $M = \max \log(\boldsymbol{A})$.
During the training phase, the values of m and M are computed over the whole data set. Because out training set contains images of land, we have m_{land} and M_{land}. However, we test the network on sea ice where the dynamic range of the images is lower.

A.3 Compressing the histogram

The reflectivity values of the sea ice and water are very low compared to land. Thus, the distribution of the pixel values of sea ice images is shifted. When the image is normalized, the distribution of the normalized image is thus shifted. The network is then forced to process an image with a different distribution, and this could lead to poor result, and maybe artifacts. We want to check the changes triggered by a shift of dynamic.
We thus *compress* the histogram of the input image to see if the artifact we have observed on the denoised sea ice images are related to the distribution of the input image.

A.3.1 Proposed experiment

Let $\alpha \in [0, 1]$ be the *compression* parameter of the input image histogram.
Let us define the stretched image $\boldsymbol{A_\alpha}$ as:

$$\boldsymbol{A_\alpha} = \alpha \ \boldsymbol{A}$$

where $\alpha \in [0, 1]$
By keeping the previous normalization, we have:

$$\begin{aligned} \boldsymbol{y_\alpha} &= \frac{\log(\alpha \ \boldsymbol{A}) - m}{M - m} \\ &= \frac{\log \boldsymbol{A} - m}{M - m} + \frac{\log \alpha}{M - m} \\ &= \boldsymbol{y} + \frac{\log \alpha}{M - m} \end{aligned}$$

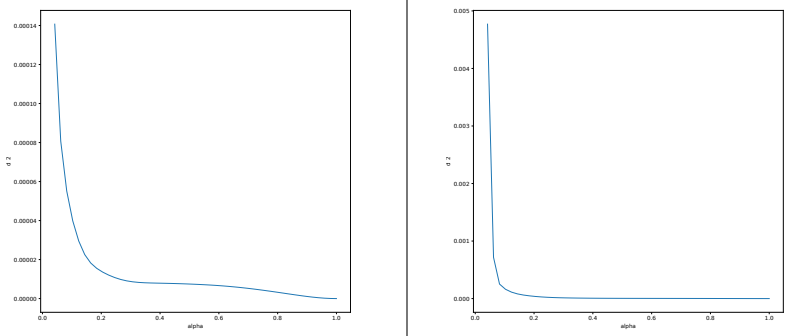
By keeping the same normalization parameters m and M, compressing the histogram of $\boldsymbol{A}$ leads to a shift of $\frac{\log \alpha}{M-m}$ for the normalized image in $[0, 1]$.
Based on $\boldsymbol{A}$, the network predict $\hat{\boldsymbol{r}}$; based on $\boldsymbol{A_\alpha}$, the network predicts $\hat{\boldsymbol{r}}_{\boldsymbol{\alpha}}$
Ideally, we would have $\hat{\boldsymbol{r}}_{\boldsymbol{\alpha}} = \alpha \ \hat{\boldsymbol{r}}$. To evaluate the impact of compressing by a factor α, we compute the following metric that we will later denote by $\boldsymbol{d}_\alpha$:

$$\boldsymbol{d}_\alpha = \left|\left| \frac{\log \hat{\boldsymbol{r}} - \log\left(\frac{\hat{\boldsymbol{r}}_{\boldsymbol{\alpha}}}{\alpha}\right)}{\log \hat{\boldsymbol{r}}} \right|\right|^2 \tag{A.1}$$

We can also use the Structural Similarity Index Measure (SSIM) to provide a metric on similarity between $\hat{\boldsymbol{r}}_{\boldsymbol{\alpha}}$ and $\alpha \, \hat{\boldsymbol{r}}$.

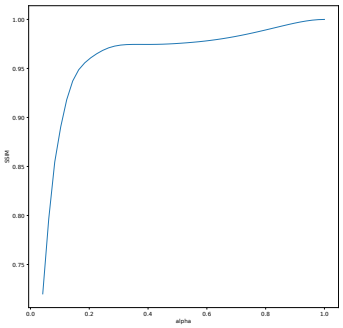
A.3.2 Experimental results

We use the weights of the network trained on the whole our data set described in 4.2.2.
The metrics are computed on 10 images of 512×512 pixels extracted from the first and the third sub-swath of one Sentinel-1 GRDM EW image.
50 values of α are used to plot the evolution of the metrics with respect to α.
Quantitative results representing d_α and the SSIM with respect to α are shown in Figure A.1.
Qualitative results on Sentinel-1 GRDM EW of river Ob in Russia are presented in Figure A.2.

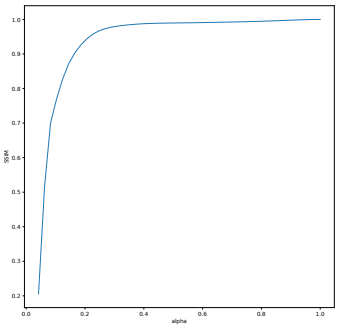


$\boldsymbol{d_2}$ w.r.t. α, HH images

$\boldsymbol{d_2}$ w.r.t. α, HV images



SSIM w.r.t. α, HH images



SSIM w.r.t. α, HV images

Figure A.1: Quantitative results: the plots represent the evolution of d_α (defined in equation A.1) with respect to α (first row) and the evolution of the SSIM with respect to α (second row). For values of α such that $\alpha \leq 0.2$, their is a drop of both metrics, meaning that the relation $\hat{\boldsymbol{r}}_{\boldsymbol{\alpha}} = \alpha\, \hat{\boldsymbol{r}}$ is not respected.

Based on the quantitative and qualitative results, we can see that the normalization affects the results of the network: when the value of α is too low, the estimated image is blurred and the details are not correctly restored... Low values of α correspond to reflectivity values of sea ice images. Thus, we need to be careful when using such normalization especially when the test data are images from areas of low reflectivities. In the training step, we can artificially change the dynamic of the input images to improve the network robustness to a shift of dynamic.

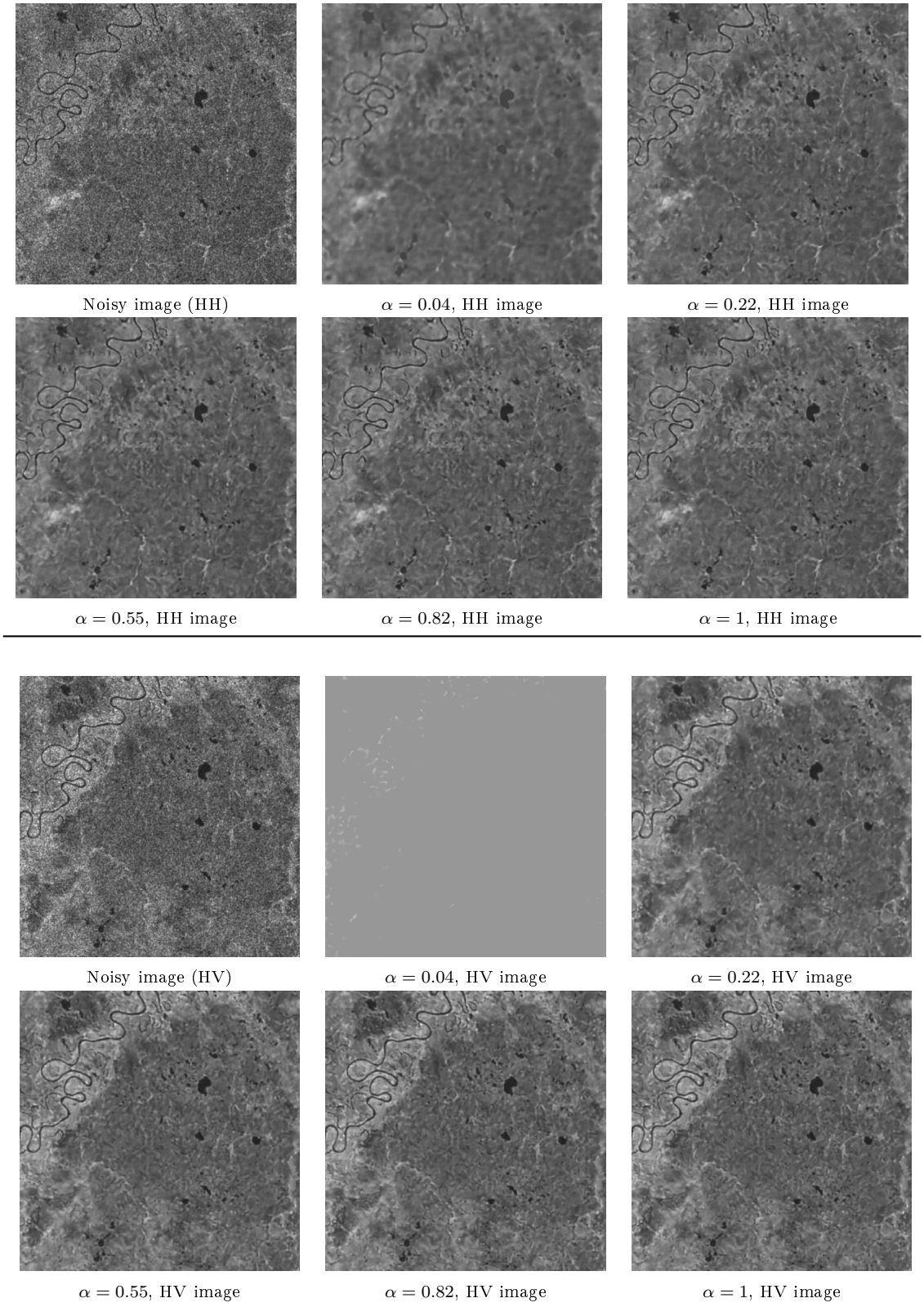


Figure A.2: Result Sentinel-1 GRDM EW images of river Ob, Russia. The patch is extracted from the first sub-swath

Bibliography

[1] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geoscience and Remote Sensing Magazine*, vol. 1, no. 1, pp. 6–43, 2013.

[2] E. Dalsasso, "Deep learning for SAR imagery : from denoising to scene understanding," Theses, Institut Polytechnique de Paris, Mar. 2022.

[3] C. Rambour, A. Budillon, A. Johnsy, L. Denis, F. Tupin, and G. Schirinzi, "From Interferometric to Tomographic Synthetic Aperture Radar. Scatterer unmixing in urban areas: A review of synthetic aperture radar tomography-processing techniques," *IEEE geoscience and remote sensing magazine*, vol. 8, no. 2, 2020.

[4] L. Cicala, C. V. Angelino, N. Fiscante, and S. L. Ullo, "Landsat-8 and sentinel-2 for fire monitoring at a local scale: A case study on vesuvius," in *2018 IEEE International Conference on Environmental Engineering (EE)*, 2018, pp. 1–6.

[5] C. Liu, F. Tupin, and Y. Gousseau, "Training CNNs on speckled optical dataset for edge detection in SAR images," *ISPRS Journal of Photogrammetry and Remote Sensing*, 2020.

[6] C. Liu, R. Abergel, Y. Gousseau, and F. Tupin, "LSDSAR, a Markovian a contrario framework for line segment detection in SAR images," *Pattern Recognition*, vol. 98, Feb. 2019.

[7] R. Abergel, L. Denis, S. Ladjal, and F. Tupin, "Subpixellic methods for sidelobes suppression and strong targets extraction in single look complex SAR images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 3, pp. 759–776, 2018.

[8] N. Gasnier, L. Denis, R. Fjørtoft, F. Liege, and F. Tupin, "LAKE DETECTION WITH SENTINEL-1 DATA USING A GRAB-CUT METHOD AND ITS MULTI-TEMPORAL EXTENSION," in *IGARSS*, Kuala Lumpur, Malaysia, 2022.

[9] G. Nicolas, L. Denis, R. Fjørtoft, F. Liege, and F. Tupin, "Narrow River Extraction from SAR Images Using Exogenous Information," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 5720–5734, 2021.

[10] A. Budillon, L. Denis, C. Rambour, G. Schirinzi, and F. Tupin, "Regularized SAR Tomography Approaches," in *IGARSS (IEEE International Geoscience and Remote Sensing Symposium)*, Hawaï, United States, Sep. 2020.

[11] C. Rambour, L. Denis, and F. Tupin, "3D Buildings Reconstruction with SAR Tomography Guided by Partial Footprints Information," in *EUSAR 2021*, VIRTUEL, Germany, Mar. 2021.

[12] C.-A. 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.

[13] ——, "NL-InSAR Nonlocal interferogram estimation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 4, pp. 1441–1452, 2010.

[14] C.-A. Deledalle, L. Denis, S. Tabti, and F. Tupin, "MuLoG, or How to apply Gaussian denoisers to multi-channel SAR speckle reduction," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4389–4403, Sep. 2017.

[15] X. Su, C.-A. Deledalle, F. Tupin, and H. Sun, "Two-step multitemporal nonlocal means for synthetic aperture radar images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 10, pp. 6181–6196, 2014.

[16] W. Zhao, C.-A. Deledalle, L. Denis, H. Maitre, J.-M. Nicolas, and F. Tupin, "Ratio-Based Multitemporal SAR Images Denoising RABASAR," *IEEE Transactions on Geoscience and Remote Sensing*, 2019.

[17] E. Dalsasso, X. Yang, L. Denis, F. Tupin, and W. Yang, "SAR Image Despeckling by Deep Neural Networks from a pre-trained model to an end-to-end training strategy," *Remote Sens.*, vol. 12, no. 16, p. 2636, 2020.

[18] 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.

[19] ——, "As if by magic self-supervised training of deep despeckling networks with merlin," *IEEE Transactions on Geoscience and Remote Sensing*, p. early access, 2021.

[20] C.-A. Deledalle, L. Denis, F. Tupin, A. Reigber, and M. Jager, "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, Apr. 2015.

[21] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise Learning Image Restoration without Clean Data," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2965–2974.

[22] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992.

[23] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[24] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1620–1630, 2013.

[25] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, 2005, pp. 60–65 vol. 2.

[26] A. Hyvarinen, E. Oja, P. Hoyer, and J. Hurri, "Image feature extraction by sparse coding and independent component analysis," in *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)*, vol. 2, 1998, pp. 1268–1273 vol.2.

[27] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering," *IEEE Trans. Imag. Proc.*, vol. 16, no. 8, pp. 2080–2095, Agu.

[28] V. Santhanam, V. I. Morariu, and L. S. Davis, "Generalized deep image to image regression," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jul 2017, pp. 5395–5405.

[29] F. Zhang, N. Cai, J. Wu, G. Cen, H. Wang, and X. Chen, "Image denoising method based on a deep convolution neural network," *IET Image Processing*, vol. 12, no. 4, pp. 485–493, 2018.

[30] K. Zhang, W. Zuo, and L. Zhang, "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, 2018.

[31] A. Krull, T.-O. Buchholz, and F. Jug, "Noise2Void - Learning Denoising From Single Noisy Images," 06 2019, pp. 2124–2132.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[33] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851.

[34] J. W. Goodman, *Speckle phenomena in optics theory and applications*. Roberts and Company Publishers, 2007.

[35] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. US Government printing office, 1968, vol. 55.

[36] J. Lee, "Digital image smoothing and the sigma filter," *Computer vision, graphics, and image processing*, vol. 24, no. 2, pp. 255–269, 1983.

[37] C.-A. Deledalle, F. Tupin, and L. Denis, "Patch similarity under non gaussian noise," in *2011 18th IEEE International Conference on Image Processing*. IEEE, Sep. 2011.

[38] C.-A. Deledalle, L. Denis, and F. Tupin, "How to compare noisy patches? patch similarity beyond gaussian noise," *International Journal of Computer Vision*, vol. 99, no. 1, pp. 86–102, Mar. 2012.

[39] 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.

[40] D. Cozzolino, S. Parrilli, G. Scarpa, G. Poggi, and L. Verdoliva, "Fast adaptive nonlocal SAR despeckling," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 2, pp. 524–528, Feb. 2014.

[41] J. Chen, Y. Chen, W. An, Y. Cui, and J. Yang, "Nonlocal filtering for polarimetric SAR data A pretest approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 5, pp. 1744–1754, 2010.

[42] C.-A. Deledalle, L. Denis, G. Poggi, F. Tupin, and L. Verdoliva, "Exploiting patch similarity for SAR image processing the nonlocal paradigm," *IEEE Sig. Proc. Mag.*, vol. 31, no. 4, pp. 69–78, 2014.

[43] L. Torres, S. J. Sant'Anna, C. da Costa Freitas, and A. C. Frery, "Speckle reduction in polarimetric SAR imagery with stochastic distances and nonlocal means," *Pattern Recognition*, vol. 47, no. 1, pp. 141–157, 2014.

[44] C.-A. Deledalle, L. Denis, F. Tupin, A. Reigber, and M. Jager, "NL-SAR a unified nonlocal framework for resolution-preserving (Pol)(In) SAR denoising," *IEEE TGRS*, vol. 53, no. 4, pp. 2021–2038, 2015.

[45] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Simul*, vol. 4, pp. 490–530, 2005.

[46] G. Aubert and J.-F. Aujol, "A variational approach to removing multiplicative noise," *SIAM Journal on Applied Mathematics*, vol. 68, no. 4, pp. 925–946, Jan. 2008.

[47] L. Denis, F. Tupin, J. Darbon, and M. Sigelle, "SAR image regularization with fast approximate discrete minimization," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1588–1600, Jul. 2009.

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

[49] G. Ferraioli, C.-A. Deledalle, L. Denis, and F. Tupin, "Parisar: Patch-based estimation and regularized inversion for multibaseline SAR interferometry," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1626–1636, Mar. 2018.

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

[51] S. Durand, J. Fadili, and M. Nikolova, "Multiplicative noise cleaning via a variational method involving curvelet coefficients," in *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer, 2009, pp. 282–294.

[52] 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)*. IEEE, 2017, pp. 5438–5441.

[53] D. Cozzolino, L. Verdoliva, G. Scarpa, and G. Poggi, "Nonlocal CNN SAR Image Despeckling," *Remote Sens.*, vol. 12, no. 6, p. 1006, 2020.

[54] G. Chierchia, M. El Gheche, G. Scarpa, and L. Verdoliva, "Multitemporal SAR Image Despeckling Based on Block-Matching and Collaborative Filtering," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 10, pp. 5467–5480, 2017.

[55] F. Lattari, B. Gonzalez Leon, F. Asaro, A. Rucci, C. Prati, and M. Matteucci, "Deep learning for SAR image despeckling," *Remote Sens.*, vol. 11, no. 13, p. 1532, 2019.

[56] S. Vitale, G. Ferraioli, and V. Pascazio, "Multi-Objective CNN-Based Algorithm for SAR Despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 11, pp. 9336–9349, 2021.

[57] P. Wang, H. Zhang, and V. M. Patel, "SAR image despeckling using a convolutional neural network," *IEEE Sig. Proces. Let.*, vol. 24, no. 12, pp. 1763–1767, 2017.

[58] ——, "Generative adversarial network-based restoration of speckled SAR images," in *2017 IEEE CAMSAP*. IEEE, 2017, pp. 1–5.

[59] Q. Zhang, Q. Yuan, J. Li, Z. Yang, and X. Ma, "Learning a dilated residual network for SAR image despeckling," *Remote Sens.*, vol. 10, no. 2, p. 196, 2018.

[60] E. Dalsasso, L. Denis, and F. Tupin, "How to handle spatial correlations in SAR despeckling Resampling strategies and deep learning approaches," in *13th European Conference on Synthetic Aperture Radar (EUSAR)*. VDE ITG, 2021, pp. 1233–1238.

[61] 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.

[62] Y. Yuan, J. Guan, P. Feng, and Y. Wu, "A Practical Solution for SAR Despeckling With Adversarial Learning Generated Speckled-to-Speckled Images," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[63] 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*, 2021.

[64] F. Sica, G. Gobbi, P. Rizzoli, and L. Bruzzone, "Phi-net: Deep residual learning for InSAR parameters estimation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 5, pp. 3917–3941, May 2021.

[65] B. Rasti, Y. Chang, E. Dalsasso, L. Denis, and P. Ghamisi, "Image restoration for remote sensing: Overview and toolbox," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 2, pp. 201–230, Jun. 2022.

[66] M. Havasi, R. Jenatton, S. Fort, J. Z. Liu, J. Snoek, B. Lakshminarayanan, A. M. Dai, and D. Tran, "Training independent subnetworks for robust prediction," in *ICLR*, 2021.

[67] E. Dalsasso, I. Meraoumia, L. Denis, and F. Tupin, "Exploiting multi-temporal information for improved speckle reduction of Sentinel-1 SAR images by deep learning," in *IGARSS 2021*, Bruxelles (virtual), Belgium, Jul. 2021.

[68] I. Meraoumia, E. Dalsasso, L. Denis, R. Abergel, and F. Tupin, "Multi-temporal speckle reduction with self-supervised deep neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, Jan. 2023.

[69] A. Korosov, D. Demchev, N. Miranda, N. Franceschi, and J.-W. Park, "Thermal denoising of cross-polarized sentinel-1 data in interferometric and extra wide swath modes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2021.

[70] S. Quegan and J. J. Yu, "Filtering of multichannel sar images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 11, pp. 2373–2379, 2001.

[71] T. T. Le, A. Atto, E. Trouvé, and J.-M. Nicolas, "TEMPORAL ADAPTIVE FILTERING OF SAR IMAGE TIME SERIES BASED ON THE DETECTION OF STABLE AND CHANGE AREAS," in *5th TerraSAR-X / 4th TanDEM-X Science Team Meeting - 2013*, Oberpfaffenhofen, Germany, Jun. 2013, p. 4 pages.

[72] ——, "Adaptive Multitemporal SAR Image Filtering Based on the Change Detection Matrix," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, p. 5 pages, Apr. 2014, 5 pages.

[73] T. T. Lê, A. M. Atto, E. Trouvé, A. Solikhin, and V. Pinel, "Change detection matrix for multitemporal filtering and change analysis of sar and polsar image time series," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 107, pp. 64–76, 2015, multitemporal remote sensing data analysis.

[74] N. Gasnier, L. Denis, and F. Tupin, "On the use and denoising of the temporal geometric mean for SAR time series," *IEEE Geoscience and Remote Sensing Letters*, 2021.

[75] N. Gasnier, E. Dalsasso, L. Denis, and F. Tupin, "Despeckling sentinel-1 grd images by deep-learning and application to narrow river segmentation," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 2021, pp. 2995–2998.

[76] I. Meraoumia, E. Dalsasso, L. Denis, and F. Tupin, "FAST STRATEGIES FOR MULTI-TEMPORAL SPECKLE REDUCTION OF SENTINEL-1 GRD IMAGES," in *IGARSS*, Kuala Lumpur, Malaysia, 2022.

[77] R. Bamler and P. Hartl, "Synthetic aperture radar interferometry," *Inverse problems*, vol. 14, no. 4, p. R1, 1998.

[78] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, "The expressive power of neural networks a view from the width," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6232–6240.

[79] 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.

[80] L. Gomez, M. E. Buemi, J. C. Jacobo-Berlles, and M. E. Mejail, "A new image quality index for objectively evaluating despeckling filtering in sar images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 3, pp. 1297–1307, 2016.

[81] X. Ma, H. Hu, and P. Wu, "A no-reference edge-preservation assessment index for sar image filters under a bayesian framework based on the ratio gradient," *Remote Sensing*, vol. 14, no. 4, 2022.

[82] J. Gawlikowski, C. Rovile Njieutcheu Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher *et al.*, "A survey of uncertainty in deep neural networks," *arXiv e-prints*, pp. arXiv–2107, 2021.

[83] A. Lyzhov, Y. Molchanova, A. Ashukha, D. Molchanov, and D. Vetrov, "Greedy policy search: A simple baseline for learnable test-time augmentation," in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, ser. Proceedings of Machine Learning Research, vol. 124. PMLR, 03–06 Aug 2020, pp. 1308–1317.

[84] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[85] M. Valdenegro-Toro, "Deep sub-ensembles for fast uncertainty estimation in image classification," 2019.

[86] ——, "Sub-ensembles for fast uncertainty estimation in neural networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, October 2023, pp. 4119–4127.

[87] Y. Wen, D. Tran, and J. Ba, "Batchensemble: an alternative approach to efficient ensemble and lifelong learning," *arXiv preprint arXiv:2002.06715*, 2020.

[88] M. Raghu, K. Blumer, R. Sayres, Z. Obermeyer, B. Kleinberg, S. Mullainathan, and J. Kleinberg, "Direct uncertainty prediction for medical second opinions," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 09–15 Jun 2019, pp. 5281–5290.

[89] L. Oala, C. Heiß, J. Macdonald, M. März, W. Samek, and G. Kutyniok, "Interval neural networks: Uncertainty scores," 2020.

[90] A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," in *Advances in Neural Information Processing Systems*, vol. 31. Curran Associates, Inc., 2018.

[91] T. Pearce, A. Brintrup, M. Zaki, and A. Neely, "High-quality prediction intervals for deep learning: A distribution-free, ensembled approach," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 80. PMLR, 10–15 Jul 2018, pp. 4075–4084.

[92] A. N. Angelopoulos, A. P. Kohli, S. Bates, M. Jordan, J. Malik, T. Alshaabi, S. Upadhyayula, and Y. Romano, "Image-to-image regression with distribution-free uncertainty quantification and applications in imaging," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 17–23 Jul 2022, pp. 717–730.

[93] M. Tassano, J. Delon, and T. Veit, "Fastdvdnet: Towards real-time deep video denoising without flow estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[94] V. Gaya, E. Dalsasso, L. Denis, F. Tupin, B. Pinel-Puysségur, and C. Guérin, "Débruitage multi-modal d'images radar à synthèse d'ouverture par apprentissage profond auto-supervisé," in *GRETSI*, Grenoble, France, Aug. 2023.

[95] D. Valsesia and E. Magli, "Permutation invariance and uncertainty in multitemporal image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2022.

[96] J. Lohse, "On Automated Classification of Sea Ice Types in SAR Imagery," PhD thesis, UiT The Arctic University of Norway, Tromsø, Norway, Mar 2021, available at https://hdl.handle.net/10037/20606.

[97] J. Lohse, A. P. Doulgeris, and W. Dierking, "Mapping sea-ice types from sentinel-1 considering the surface-type dependent effect of incidence angle," *Annals of Glaciology*, vol. 61, no. 83, pp. 260–270, 2020.

[98] J. Lohse, C. Taelman, A. Everett, and N. Hughes, "Fully-automated navigation support for vessels in the Arctic: An application and validation example of ice type mapping during the CIRFA cruise 2022," Mar 2023.

[99] I. Meraoumia, E. Dalsasso, L. Denis, and F. Tupin, "Débruitage multi-temporel d'images radar à synthèse d'ouverture par apprentissage profond auto-supervisé," in *GRETSI 2022*, Nancy, France, Sep. 2022.

ECOLE
DOCTORALE

Titre : Apprentissage profond pour l'interprétation des images satellitaires

Mots clés : SAR, apprentissage profond, imagerie satellitaire

Résumé : Le radar à synthèse d'ouverture (SAR) n'est pas impacté par la présence de nuages ou la luminosité et permet donc l'acquisition d'images riches en informations pour l'observation de la Terre (chapitre 1). De fortes fluctuations appelées "speckle" sont néanmoins visibles sur ces images et rendent leur interprétation difficile. Le "speckle" est un phénomène intrinsèque à l'illumination cohérente de la scène par le capteur : des images sans speckle ne peuvent donc pas être acquises. Les propriétés du speckle sont différentes de celles du bruit additif blanc gaussien usuellement utilisé en imagerie optique. Les algorithmes de despeckling sont donc propres aux statistiques du speckle du modèle de Goodman (chapitre 2). Récemment, des méthodes d'apprentissage profond ont donné de très bons résultats pour la restauration d'une seule image SAR. Les travaux proposés utilisent le traitement conjoint de plusieurs images SAR pour améliorer leur restauration en exploitant l'information commune tout en empêchant la propagation de potentielles différences (chapitre 3).

Le chapitre 4 est centré sur le despeckling des images Sentinel-1 GRDM Extra Wide de la glace de mer. La glace se déplaçant rapidement sur la mer, des changements structurels apparaissent rapidement sur une zone d'intérêt, rendant les piles multi-temporelles inexploitables. Le bruit thermique de ces images ne peut pas être négligé car les valeurs de réflectivité de l'eau et de la glace sont très faibles et proches du seuil du bruit thermique. Notre méthode de despeckling utilise et restaure conjointement les canaux polarimétriques HH et HV. L'entrainement autosupervisé du réseau s'inspire de la méthode SAR2SAR, en prenant en entrée des images corrigées où la composante de bruit thermique a été supprimée. La qualité des images Sentinel-1 de l'Arctique restaurées avec notre approche est bien meilleure que celle obtenue avec d'autres techniques de restauration.

Utiliser l'information partagée au sein d'une pile multi-temporelle tout en ignorant l'impact des changements temporels améliore le despeckling. Des méthodes de despeckling multitemporel basées sur un moyennage temporel ou l'utilisation d'une super-image sont d'abord présentées dans le chapitre 5. Un modèle génératif est ensuite proposé afin d'expliciter la formation d'une pile multi-temporelle d'images SAR en tenant compte des corrélations spatiales et temporelles du speckle. Une extension multitemporelle de la méthode MERLIN est basée sur ce modèle génératif et prend en entrée des images additionnelles de la même zone mais acquises à des dates différentes. L'entrainement du réseau est non supervisé et s'inspire de la méthode Noise2Noise : la partie réelle (ou la partie imaginaire) de l'image et les dates additionnelles sont transmises au réseau et la partie imaginaire (ou la partie réelle) est utilisée comme cible. L'ajout d'images supplémentaires améliore la restauration des images SAR avec un gain décroissant. Un blanchiment temporel est proposé pour éviter une perte de performance liée aux corrélations temporelles entre les canaux d'entrée.

L'absence d'image de référence rend l'évaluation des méthodes de despeckling difficile. Le chapitre 6 se concentre sur la quantification des incertitudes liées à la prédiction d'un réseau. Des travaux combinant le despeckling et l'estimation d'une carte d'incertitudes sont d'abord présentés. Dans le cadre d'origine, une seule valeur de réflectivité est prédite pour chaque pixel, alors que nous visons à prédire une distribution pour chaque pixel. Les paramètres des lois uniforme puis inverse gamma sont estimés lors de l'entraînement. La difficulté à quantifier les incertitudes dans un cadre d'apprentissage auto-supervisé où le niveau de bruit est élevé est ensuite discutée. En utilisant le réseau MERLIN, la prédiction de la carte moyenne des différences entre les prédictions basées sur la partie réelle et imaginaire fournit une carte d'incertitudes satisfaisante.

Title : Deep learning for remote sensing images and their interpretation

Keywords : SAR, deep learning, remote sensing

Abstract : Synthetic Aperture Radar (SAR) images are not affected by the presence of clouds or variations of sunlight. They provide very useful information for Earth observation (chapter 1). They are impacted by strong fluctuations called ”speckle” which make their interpretation difficult. The speckle is a phenomenon intrinsic to the coherent illumination of the scene by the radar, meaning that speckle-free images can not be captured and used as reference to train models. The properties of speckle are different from that of the traditional additive white Gaussian noise used to model corruptions in optical images, and proper despeckling algorithms are needed. Most of them rely on statistics derived from the Goodman's model (chapter 2). Recently, deep learning based methods have been very successful at despeckling a single SAR image. This work focuses on improving the despeckling performance by jointly processing several input images to exploit the common information while still preventing the propagation of differences from one image to another (chapter 3).

The despeckling of Sentinel-1 GRDM Extra Wide images of sea ice is studied in Chapter 4 for sea ice studies. The ice is shifting quickly on the sea and multi-temporal stacks of a specific area can not be used for despeckling purposes due to structural changes. In the images, thermal noise can not be neglected because the reflectivity values of water and ice are very low and close to the thermal noise floor. We propose a dual-polarimetric despeckling framework where HH and HV polarimetric channels are used as input and are jointly despeckled in a single pass. The network is trained in a self-supervised way inspired by the existing SAR2SAR framework and takes corrected images where the thermal noise floor level has been removed as input. Our approach shows a clear improvement over existing image restoration techniques on Sentinel-1 images of the Artic.

Despeckling can be improved by combining measurements pertaining to common information within the temporal stack while ignoring data impacted by temporal changes. First, multi-temporal despeckling methods using temporal averaging and the computation of a super-image (i.e. despeckled temporal mean image) are introduced at the beginning of Chapter 5. A generative model is then proposed to explicit the statistics of SAR multi-temporal stacks and account for the spatial and temporal correlations of speckle. A multi-temporal extension of the existing MERLIN framework is derived from this model. The network is fed with additional images of the same area acquired at different dates. It is trained in an unsupervised way inspired by the Noise2Noise framework: the real part (or the imaginary part) of the image and additional dates are fed to the network and the imaginary part (or the real part) is used as a target. Adding more images continuously improves the despeckling performance, but with diminishing gains. A temporal whitening is proposed to prevent the drop of performance of the network when the input channels are temporally correlated.

Despeckling methods are hard to evaluate because of the lack of ground truth images. Chapter 6 focuses on uncertainty quantification for despeckling using deep learning. First, works are presented to combine despeckling and estimation of the uncertainty map during the training. Starting from a framework where only one value is predicted for each pixel, we aim at predicting a distribution for each pixel. Parameters of uniform and inverse gamma distributions are estimated. The sharper the distribution, the more certain the network is of its prediction. We discuss the difficulty of estimating uncertainties in a self-supervised learning framework where the noise level is high and the limits faced by our formulations. Working with the MERLIN framework, an estimation of an uncertainty map is proposed based on the expected difference map between predictions from the real and imaginary parts.

Institut Polytechnique de Paris
91120 Palaiseau, France