Learning a versatile representation of SAR data for regression and segmentation by leveraging self-supervised despeckling with MERLIN

Emanuele Dalsasso, Clément Rambour, Loïc Denis, Florence Tupin

▶ **To cite this version:**

Emanuele Dalsasso, Clément Rambour, Loïc Denis, Florence Tupin. Learning a versatile representation of SAR data for regression and segmentation by leveraging self-supervised despeckling with MERLIN. 2023. hal-04245654

HAL Id: hal-04245654

https://telecom-paris.hal.science/hal-04245654

Preprint submitted on 17 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning a versatile representation of SAR data for regression and segmentation by leveraging self-supervised despeckling with MERLIN

Emanuele Dalsasso[a], Clément Rambour[a], Loïc Denis[b,c], and Florence Tupin[b]
[a]CÉDRIC, Conservatoire National des Arts et Métiers, Paris, France
[b]LTCI, Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France
[c]Université Jean Monnet Saint-Etienne, CNRS, Institut d'Optique Graduate School, Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Étienne

Abstract

Synthetic Aperture Radar (SAR) images are abundantly available, yet labels are often missing. Thus, training a neural network in a fully supervised manner is arduous. In this work, we leverage MERLIN, a self-supervised despeckling algorithm, to learn a mapping of SAR images into a representation space shared among despeckling, segmentation and regression. Our experiments demonstrate that the joint training of a neural network for these three tasks reduces considerably the need for labeled data to solve the supervised tasks.

1 Introduction

Deep learning led to unprecedented results in many fields, among which remote sensing. A wide number of applications related to Earth Observation from satellite data are nowadays based of the use of deep neural networks [1]. In a classical supervised deep learning framework, a model can learn a given task by looking at a set of annotated examples. If the training dataset is big enough, the model can generalize to unseen examples. However, supervised training strategies are limited by the availability of labeled data. In particular, remote sensing is characterized by a wide availability of raw data, but labels are often scarce. This issue can be mitigated by resorting to self-supervised learning (SSL).

Instead of training a neural network on a big set of annotated samples for a specific task, an SSL framework leverages unlabeled samples to learn a meaningful representation of the data that can be transferred on downstream tasks only by looking at a reduced set of annotated examples [2]. To extract a data structure encoding its semantic without supervision, one can resort to different techniques such as unsupervised pretext tasks or strong data augmentations, which must be carefully designed to avoid introducing unwanted properties in the network.

SSL has been successfully applied to natural images and has sparked a great interest in the remote sensing community, showing to transfer well to optical images [2, 3]. Instead, Synthetic Aperture Radar (SAR) images are considerably different from natural images in several ways. The side-looking geometry introduces specific distortions that depend on sensor orientation: not only this limits the use of techniques such as image flipping or rotation, but it makes it difficult to exploit multi-modal SSL techniques [4] for which a perfect co-registration among the different modalities is needed to solve pixel-level tasks. Moreover, as SAR images are acquired by a coherent system, they are characterized by the presence of the speckle phenomenon, yielding specific statistics. Thus, the use of SSL to learn a representation of SAR images is still at its infancy.

In this paper, we propose a multi-task weakly-supervised framework for SAR images that leverages self-supervised despeckling. Inspired from DenoiSeg [5], where a neural network is jointly trained on self-supervised denoising and supervised segmentation, we show that the despeckling tasks helps the network to learn a versatile representation of SAR data that transfers well to regression and segmentation. Moreover, our experiments demonstrate that the use of despeckling alleviates the need for labeled data in supervised downstream tasks.

2 Method

In the proposed weakly supervised framework, a neural network is jointly trained for despeckling, segmentation and regression. The network is composed of a common backbone that learns a representation of SAR data that is shared between the three tasks. Then, three task-specific heads are tuned to exploit the common information to produce the restored SAR image, the segmentation map and the regression map. As despeckling is self-supervised, during training the network always sees all the dataset, making it possible to learn a rich representation shared by all the tasks, whereas the parameters of the two supervised downstream-specific tasks are learned only based on the labels that are available. In this way, while the network learns to suppress speckle from SAR images, it co-learns to infer the segmentation map and the regression map. The proposed training pipeline shows that despeckling borrows itself really well to learn a versatile representation of SAR data that can be shared among multiple tasks. The rest of the section recalls the principle of self-supervised despeck-

ling with MERLIN [6] and describes the losses used for the chosen segmentation and regression tasks.

2.1 SAR despeckling with MERLIN

The random phasor describing a Single-Look Complex SAR measurement is described by Goodman [7] as the random walk sum of N elementary scatterers, yielding $z = Ae^{j\phi} = \sum_{n=0}^{N} A_n e^{j\phi_n}$, with A the amplitude of the resultant phasor and ϕ its phase. Provided that the N elementary scatterers are independent and identically distributed, if N is large the central limit theorem follows. Thus, we have that the signal z measured for a homogeneous area characterized by a reflectivity R follows a circular Gaussian distribution:

$$\mathrm{p}_Z(z) = \frac{1}{\pi R} \exp\left(-\frac{|z|^2}{R}\right). \tag{1}$$

It can be demonstrated that, under the hypothesis stated above, the real and imaginary parts a and b of the complex amplitude $z = x + jy$ are uncorrelated and, as they follow a Gaussian distribution, this leads to their independence:

$$\mathrm{p}_Z(z) = \mathrm{p}_X(x)\mathrm{p}_Y(y) = \frac{1}{\pi R} \exp\left(-\frac{x^2+y^2}{R}\right). \tag{2}$$

MERLIN [8] leverages the independence of real and imaginary parts to decompose the SLC image $\mathbf{z}$ into two independent sub-images $\mathbf{x}$ and $\mathbf{y}$, one serving as input to a neural network defined as f_θ and the other one supervises the training. By denoting with $\tilde{\boldsymbol{R}}^x = f_\theta(\boldsymbol{x})$ the estimation of the reflectivity given the real part in input, the loss function for the despeckling task can be computed as follows:

$$\begin{aligned}\mathcal{L}_{\mathrm{D}}(\tilde{\boldsymbol{R}}^x, \boldsymbol{y}) &= \frac{1}{K}\sum_k -\log \mathrm{p}(y_k|\tilde{R}^x_k) \\ &= \frac{1}{K}\sum_k \frac{1}{2}\log\left(\tilde{R}^x_k\right) + \frac{y_k^2}{\tilde{R}^x_k}, \end{aligned} \tag{3}$$

where k indicates the k-th pixel of the image. As the role of $\boldsymbol{x}$ and $\boldsymbol{y}$ is symmetrical, they can be swapped during training. At inference time, the estimated reflectivity $\tilde{\boldsymbol{R}}$ is given by:

$$\tilde{\boldsymbol{R}} = \frac{f_\theta(\boldsymbol{x}) + f_\theta(\boldsymbol{y})}{2} = \frac{\tilde{\boldsymbol{R}}^x + \tilde{\boldsymbol{R}}^y}{2} \tag{4}$$

2.2 MERLIN-multitask: a weakly-supervised framework for joint despeckling, segmentation and regression

We propose a multi-task neural network that makes the best use of self-supervised despeckling to learn a versatile representation of SAR data. Our work stems from the hypothesis that, in order to produce a speckle-free image, a neural network encodes the semantic and the geometry of the input SAR image. To demonstrate that the representation learned through despeckling is capable of covering a set of tasks, we study the behavior of the network on two different pixel-level tasks: building extraction (segmentation) and height estimation (regression).

2.2.1 Semantic segmentation

Our neural network is trained on the semantic task of building extraction. The problem is expressed as a binary segmentation task: each pixel is associated to a positive label if it belongs to a building, to a negative label otherwise. Thus, the weighted Binary Cross-Entropy loss is used for this task:

$$\mathcal{L}_{\mathrm{S}}(\tilde{\boldsymbol{q}}, \boldsymbol{q}) = \frac{1}{K}\sum_k w q_k \log(\tilde{q}_k) + (1-q_k)\log(1-\tilde{q}_k), \tag{5}$$

with q_k and $\tilde{q}_k$ the groundtruth and predicted segmentation mask at pixel k and w a scaling factor accounting for imbalance between positive and negative pixels in the training set.

2.2.2 Regression

The regression task that we chose is height estimation. For this purpose, weights of the regression head are optimized to minimize the Mean Squared Error (MSE) term between the groundtruth $\boldsymbol{h}$ and the estimated height $\tilde{\boldsymbol{h}}$:

$$\mathcal{L}_{\mathrm{R}}(\tilde{\boldsymbol{h}}, \boldsymbol{h}) = \frac{1}{K}\sum_k (h_k - \tilde{h}_k)^2. \tag{6}$$

2.2.3 MERLIN-multitask

In our weakly-supervised framework, the model is trained simultaneously for three tasks: segmentation, regression and despeckling. As it is an extension of MERLIN for the joint training of three tasks, we refer to it as MERLIN-multitask. When the input SAR image contains segmentation and regression annotations, the model is jointly optimized for the three tasks and to minimize the following multi-task loss:

$$\mathcal{L}_{\text{multitask}} = \lambda_S \mathcal{L}_{\mathrm{S}} + \lambda_R \mathcal{L}_{\mathrm{R}} + \lambda_D \mathcal{L}_{\mathrm{D}}. \tag{7}$$

where λ_S, λ_R and λ_D are three non-negative hyperparameters balancing the weight of the three tasks and have been chosen empirically. When the input image is not annotated, only the despeckling loss is evaluated, *i.e.* $\mathcal{L}_{\mathrm{S}}$ and $\mathcal{L}_{\mathrm{R}}$ are set to 0, and the downstream task-specific parameters are not tuned. Instead, cross-task parameters responsible for learning a mapping of the input image into the feature space are always optimized thanks to the self-supervised despeckling task, indirectly benefiting the two supervised tasks.

3 Experiments

3.1 Experimental setup

The model backbone is composed by the same U-Net architecture used in MERLIN [8]. As in [6], the segmentation head is composed by 3 convolutional layers. The first and the second layer of the segmentation head comprise respectively 64 and 32 filters of size 3×3, followed by Leaky ReLU activations. The last layer is a single $1-$d filter with a sigmoid activation that produces class probabilities for

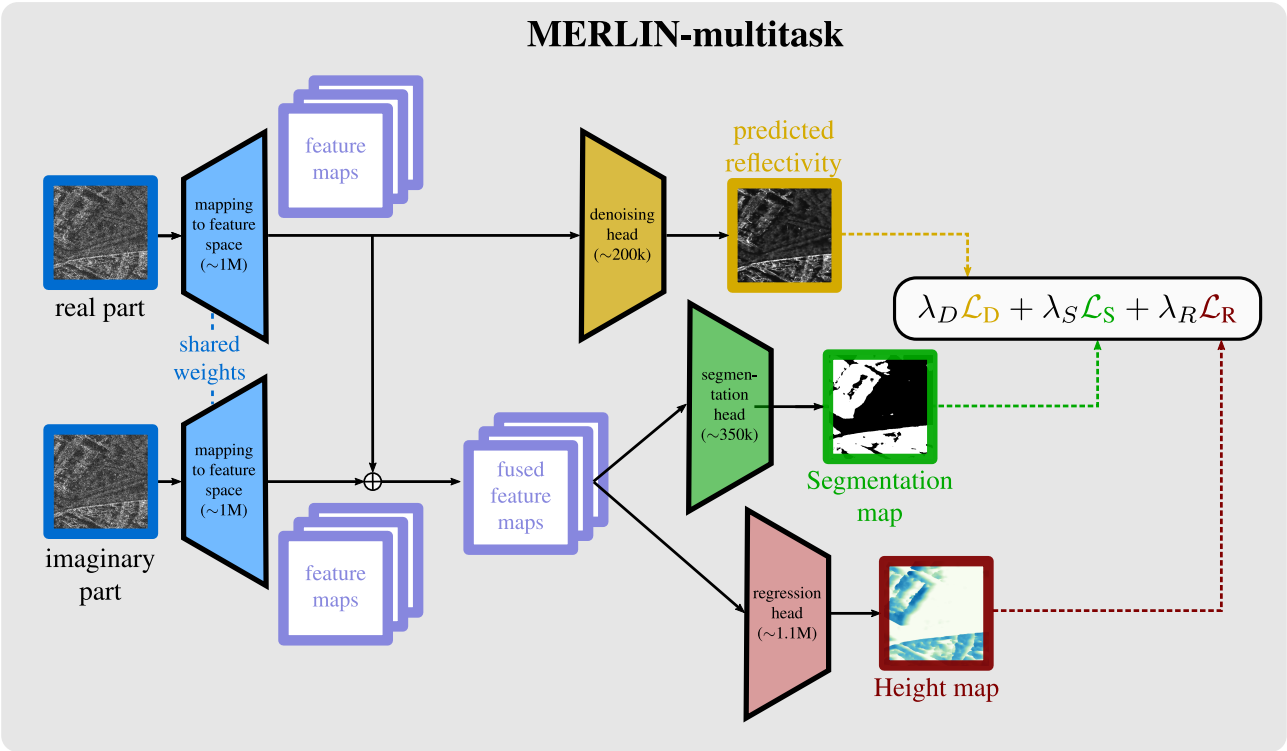

Figure 1 The proposed SSL framework for joint segmentation and regression leveraging the representation learned through self-supervised despeckling using MERLIN. A common backbone learns an encoding of the data mainly thanks to the despeckling task. The task-specific heads for segmentation and regression are lightweight (the parameters of each block are given between brackets): they resort on the representation learned through despeckling and can be tuned on a small amount of task-specific labeled data.

each image pixel. The regression head mimics the behavior of the IM2HEIGHT network [9] and is composed of 3 residual blocks.

To serve as a segmentation baseline, we chose a U-Net architecture following the same structure as the backbone composing MERLIN-multitask, while for height estimation, we used the IM2HEIGHT network proposed in [9]. It is worth to point out that the proposed MERLIN-multitask architecture has approximately 2.7M parameters (Table 1), among which $\sim$ 350k for the segmentation head and $\sim$ 1.1M for the regression head, while $\sim$ 1M of parameters are shared among the different tasks (Figure 1). Thus, both the segmentation head and the regression head are considerably lightweight compared to their baselines (U-Net has $\sim$ 1M parameters and IM2HEIGHT has $\sim$ 11.2M parameters, see Table 1), thus reducing the amount of labeled data needed to tune them.

In all our experiments, all models are trained from scratch with the set of hyperparameters indicated in Table 1. Our dataset is composed of a 6000×10000 pixels TerraSAR-X HighRes SpotLight acquisition over the city of Paris, France. The groundtruth is extracted from the BDtopo database from the french National Geographic Institute (IGN). The BDtopo is available under an open access license and it provides geo-referenced vectors describing the 2.5D building geometry. We extracted building segmentation and building height from it. In order to use them as groundtruth for the segmentation and regression tasks, respectively, annotations are projected in the slant-range geometry using the parameters contained in the metadata of the TerraSAR-X product.

The scene is separated into three non-overlapping areas for training, validation and test. In all our experiments, we consider the same training set and decompose it into smaller patches, as indicated in Table 1. It is worth to point out that the number of training patches indicated on the table corresponds to the full (100%) annotated training dataset. In our experiments with a reduced number of annotated patches, baseline models only see the patches for which labels are available. Instead, MERLIN-multitask always sees all available patches to train the feature extractor and the despeckling head, while the network co-learns to regress height and segment buildings on the portion of labeled patches.

3.2 Results

For quantitative comparison, segmentation performances are evaluated in terms of *IoU*, *F1 score* and *Accuracy*, while regression is assessed in terms of the following

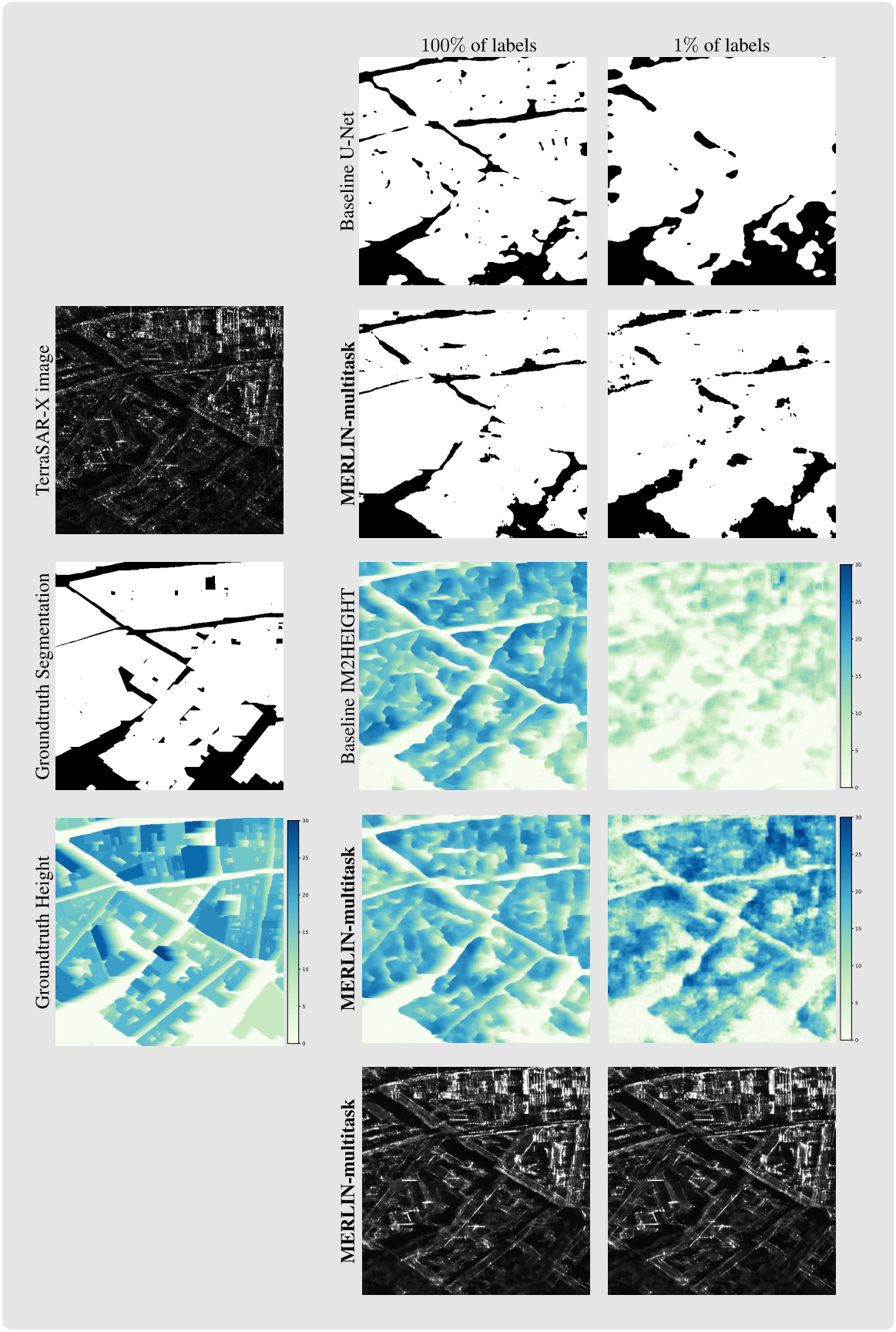

Figure 2 Comparison on building segmentation and height estimation of the proposed MERLIN-multitask with a U-Net and IM2HEIGHT. MERLIN-multitask shows that the quality of the result is much less affected by a reduction of the groundtruth labels by a factor 100 than the baseline methods.

Table 1 Description of the training parameters for all experiments carried out with a modified IM2HEIGHT architecture.

	U-Net baseline	IM2HEIGHT baseline	MERLIN-multitask
# parameters	$\sim$1M	$\sim$ 11.2M	$\sim$ 2.7M
# training patches	1189	1493	1189
patch size	512×512	256×256	512×512
stride size	128	128	128
batch size	12	1	1
# epochs	100	100	100
learning rate	10^{-2}	10^{-2}	10^{-2}
	10^{-3} after 20 epochs	10^{-3} after 30 epochs	10^{-3} after 10 epochs
	10^{-4} after 80 epochs	10^{-4} after 50 epochs	10^{-4} after 50 epochs
loss	$\mathcal{L}_S$	$\mathcal{L}_R$	$\lambda_S \mathcal{L}_S + \lambda_R \mathcal{L}_R + \lambda_D \mathcal{L}_D$
λ_S	✗	✗	0.9
λ_R	✗	✗	0.1
λ_D	✗	✗	0.9
# validation patches	5	5	5
# test patches	70	70	70

scores:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_i (h_i - \tilde{h}_i)^2} \tag{8}$$

$$\text{logRMSE} = \sqrt{\frac{1}{N}\sum_i |\log_{10}(h_i+1) - \log_{10}(\tilde{h}_i+1)|^2} \tag{9}$$

$$\text{Rel} = \frac{1}{N}\sum_i \frac{|h_i - \tilde{h}_i|}{|h_i|+1} \tag{10}$$

$$\text{Rel}_{\log} = \frac{1}{N}\sum_i \frac{|\log_{10}(h_i+1) - \log_{10}(\tilde{h}_i+1)|}{\log_{10}(h_i+1)+1} \tag{11}$$

Quantitative results are summarized in Figure 3. The higher the three segmentation scores, the better the results. As for height estimation, the lower are the errors, the better is the estimation. For both segmentation and regression, MERLIN-multitask shows a slight improvement with respect to the baselines. The more we reduce the number of annotated samples in the dataset, the higher the discrepancy between MERLIN-multitask and the baseline models. While the baseline models exploit only labeled data and are trained with only 18 batches when 1% of the annotated dataset is available, MERLIN-multitask is always fed with all SAR patches. Indeed, for the speckle reduction task, labels are not needed. Thus, the network learns a mapping from the input SAR image to the feature space that is shared with the segmentation and regression heads. While their parameters are tuned only on labeled data, the input representation is more expressive as the network has learned to model SAR data through speckle reduction. This property alleviates the need for annotated samples.

Figure 2 shows the building mask, the estimated height and the restored speckle-free SAR image produced by MERLIN-multitask. A comparison with U-Net and IM2HEIGHT is given. When all the dataset is annotated, the segmented buildings and estimated heights are close to the groundtruth for all models, although speckle reduction seems to introduce an implicit regularization and shapes of the buildings are better respected. However, when only 1% of annotated patches are available, both baselines perform poorly, showing a big gap with the result produced with the same model when trained with more labeled data. Instead, MERLIN-multitask suffers less from the scarcity of labels in the dataset and produces satisfying results both for building segmentation and height estimation.

Although speckle reduction only serves as a *pretext task* in our framework and we are mostly interested in the performances on the two supervised downstream tasks, the speckle-free images estimated with MERLIN-multitask displayed in Figure 2 show excellent restoration quality.

3.3 Ablation study

We conducted a study to disentangle the performances on segmentation and regression from the particular architecture chosen for MERLIN-multitask and show that the improvements on the downstream tasks are to be attributed to the representation extracted by the network on despeckling. To this purpose, MERLIN-multitask has been trained with different combinations of the three tasks when 1% of annotated data are available. Table 2 allows to conclude that there exists a cooperation between regression and segmentation, as there is a slight improvement in the model trained on both regression and segmentation (3^{rd} line) over the models trained on a single task (1^{st} and 2^{nd} lines). A significant improvement is observed when the two downstream tasks are trained in combination with despeckling (last three lines).

4 Conclusion

Self-supervised despeckling offers a unique opportunity to exploit the abundance of SAR data to extract its semantic and geometry. In MERLIN-multitask, we exploit the representation learned thanks to despeckling to reduce the need for labeled data in supervised downstream tasks by jointly training the network for despeckling, segmentation and re-

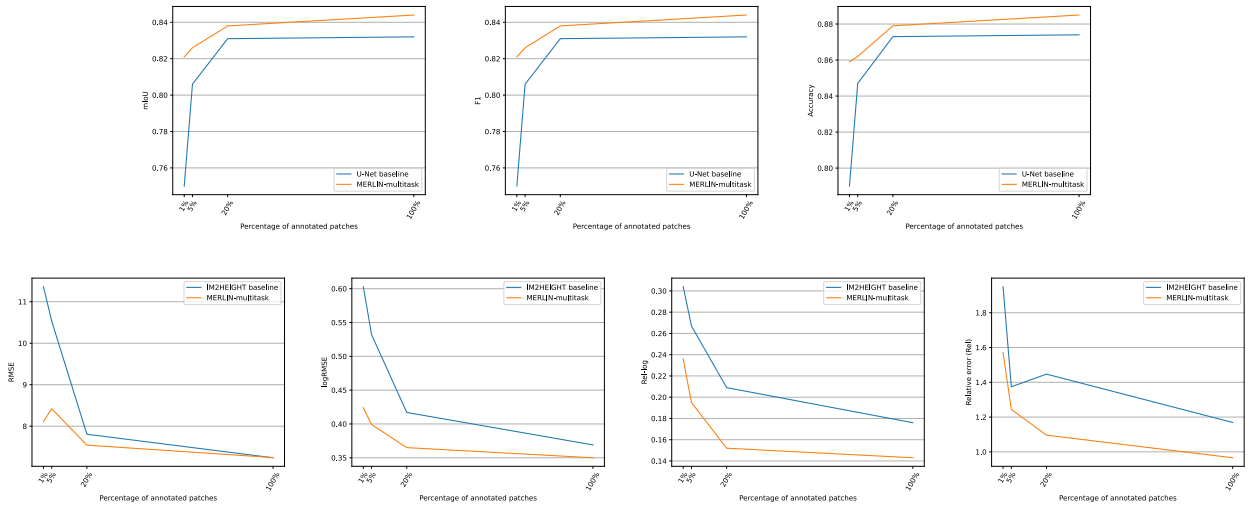

Figure 3 Quantitative evaluation of the results. Top row compares the segmentation performances of MERLIN-multitask with the U-Net baseline and demonstrate that higher scores are obtained by MERLIN-multitask, which does not suffer from data scarcity as much as the baseline. The same conclusion can be drown when observing the bottom row, comparing the errors on height estimation of the IM2HEIGHT baseline and the proposed MERLIN-multitask approach.

Table 2 Results with 1% of annotated data on different combinations of tasks with MERLIN-multitask.

λ_S	λ_R	λ_D	logRMSE (↓)	F1 (↑)
✗	✓	✗	0.482	✗
✓	✗	✗	✗	0.738
✓	✓	✗	0.480	0.775
✓	✗	✓	✗	**0.838**
✗	✓	✓	0.447	✗
✓	✓	✓	**0.424**	0.821

gression. Our experiments show that good quality can be obtained even when only few annotated samples are available, demonstrating that despeckling allows to train models with weak supervision.

The remote sensing community has seen a growing interest towards multi-modal transformers pre-trained in a self-supervised way [10]. Such models generally perform poorly on pixel-level tasks when only the SAR modality is available, as they are often pre-trained on image-level (or patch-level) tasks and the alignment between SAR and other modalities is not perfect. A promising direction is the development of multi-modal pre-trained models exploiting SAR despeckling to improve on pixel-level tasks on SAR images.

5 Acknowledgements

This project has been funded by ANR (the French National Research Agency) and DGA (Direction Générale de l'Armement) under ASTRAL project ANR-21-ASTR-0011.

6 Literature

[1] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE GRSM*, vol. 5, no. 4, pp. 8–36, 2017.

[2] Y. Wang, C. Albrecht, N. A. A. Braham, L. Mou, and X. Zhu, "Self-supervised learning in remote sensing: A review," *IEEE GRSM*, 2022.

[3] J. Prexl and M. Schmitt, "Multi-modal multi-objective contrastive learning for sentinel-1/2 imagery," in *CVPR*, 2023, pp. 2135–2143.

[4] R. Bachmann, D. Mizrahi, A. Atanov, and A. Zamir, "Multimae: Multi-modal multi-task masked autoencoders," in *ECCV*. Springer, 2022, pp. 348–367.

[5] T.-O. Buchholz, M. Prakash, D. Schmidt, A. Krull, and F. Jug, "Denoiseg: joint denoising and segmentation," in *ECCV*. Springer, 2020, pp. 324–337.

[6] E. Dalsasso, C. Rambour, N. Trouvé, and N. Thome, "Merlin-seg: self-supervised despeckling for label-efficient semantic segmentation," *hal preprint hal-04163624*, 2023.

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

[8] E. Dalsasso, L. Denis, and F. Tupin, "As if by magic: self-supervised training of deep despeckling networks with merlin," *IEEE TGRS*, vol. 60, pp. 1–13, 2021.

[9] M. Recla and M. Schmitt, "Deep-learning-based single-image height reconstruction from very-high-resolution SAR intensity data," *ISPRS*, 2022.

[10] Y. Chen, M. Zhao, and L. Bruzzone, "Incomplete multimodal learning for remote sensing data fusion," *arXiv preprint arXiv:2304.11381*, 2023.