Snow avalanche segmentation in SAR images with Fully Convolutional Neural Networks.

Filippo Maria Bianchi*, Jakob Grahn, Markus Eckerstorfer, Eirik Malnes, Hannah Vickers

Abstract—Knowledge about frequency and location of snow avalanche activity is important for forecasting and mapping of snow avalanche hazard. Traditional field monitoring of avalanche activity has limitations, especially when surveying large and remote areas. In recent years, avalanche detection in Sentinel-1 radar satellite imagery has been developed to overcome this monitoring problem. Current state-of-the-art detection algorithms, based on radar signal processing techniques, have highly varying accuracy that is on average much lower than the accuracy of visual detections from human experts. To reduce this gap, we propose a deep learning architecture for detecting avalanches in Sentinel-1 radar images. We trained a neural network on 6345 manually labelled avalanches from 117 Sentinel-1 images, each one consisting of six channels with backscatter and topographical information. Then, we tested the best network configuration on one additional SAR image. Comparing to the manual labelling (the gold standard), we achieved an F1 score above 66%, while the state-of-the-art detection algorithm produced an F1 score of 38%. A visual interpretation of the network’s results shows that it only fails to detect small avalanches, while it manages to detect some that were not labelled by the human expert.

Index Terms—Deep Learning; Saliency Segmentation; Convolutional Neural Networks; Snow Avalanches; SAR; Sentinel-1.

I. INTRODUCTION

Knowledge about the spatio-temporal magnitude of snow avalanche (hereafter avalanche) activity in a given region is important for avalanche forecasting and hazard mapping. Since traditional field monitoring has limitations, especially when surveying large and remote areas, recent approaches perform avalanche detection from synthetic aperture radar (SAR) data, which are unaffected by clouds and light conditions and allow for monitoring of large regions [1].

While an experienced operator is able to identify avalanches in SAR change detection composites (showing temporal radar backscatter change) with high confidence, automatic signal processing methods based on radar backscatter thresholding and segmentation can fail and produce a large amount of false alarms due to the highly dynamic nature of snow in the SAR images [2].

The application in remote sensing of deep learning models, such as convolutional neural networks (CNNs), outperformed previous signal processing techniques and sets the new state-of-the-art in several tasks [3]. Prominent examples are terrain surface classification [4, 5], categorization of aerial scenes [6], detection of changes in the terrain over time from SAR and optical satellite sensors [7, 8], and segmentation of objects from airborne images [9]. However, few research efforts have been devoted so far to the detection of avalanche activity from SAR data, which remains an open and challenging endeavour.

In our previous work [10], we proposed a deep learning architecture to perform binary classification of avalanches in Northern Norway. In particular, we exploited CNN models to classify windows of fixed size in two classes: 1 if the patch contains at least one avalanche, or 0 otherwise. The same approach has also been adopted for SAR-borne avalanche detection in the Alps [11] and in Norway [12]. The principal limitation of this binary avalanche classification approach is that the results are heavily influenced by the chosen window size, which makes it difficult to quantify the detection performance. If the window size is too large, it is more likely to correctly predict the presence of at least one avalanche in the image, but the resolution of the detection is too coarse and less useful. Additionally, it is not possible to distinguish between a patch that contains only one or many avalanches.

In this work, we approach the avalanche detection problem as a segmentation task, where the classification is not done at patch level, but rather at the individual pixel level. We adopt a Fully Convolutional Network (FCN) architecture, which yields for each input image a segmentation mask of the same size. The FCN model is independent of the input size and, once trained, can be seamlessly applied to larger images. Our approach solves the drawbacks related to the dependency on the window size, making the evaluation of the detection accuracy more precise and reliable. By detecting avalanches at pixel level it is possible to determine the exact location of each avalanche, thus addressing the vagueness in the results yielded by the patch-based classification.

To improve the detection accuracy, we exploit exogenous topographical information computed from a digital elevation model (DEM), including a new feature that we call potential angle of reach (PAR). The motivation is to leverage information about terrain that is likely to produce avalanches.

II. DATASET

Our dataset consists of 118 Sentinel-1 (S1) scenes from five different orbits, covering two mountainous regions in Northern Norway in the period Oct 2014–April 2017. The SAR data was geocoded, terrain corrected and calibrated to provide radar backscatter on a rectangular grid (WGS-84,UTM z33N). Each S1 scene consists of an activity and a reference SAR image of similar geometry (asc or desc) and orbit (e.g. 168), acquired 6 or 12 days (before 2015) apart. The S1 scenes have an approximate size of 11.500 × 5.500 pixels, and each pixel covers 20 × 20 meters. To remove noise and restrict the range of the backscatter
Fig. 1. In (a, b) the difference in the VV and VH channels from the input SAR data. In (e), the product VVVH of the squared differences. In (d, e), the slope and the PAR feature maps. Only a small area (1k x 1k pixels) of the actual scene is depicted.

to intervals where avalanches are visible, we clipped the values in the SAR images to [-25dB, -5dB]. In each scene, a human expert constructed a binary classification mask that indicates whether a pixel in the scene is an avalanche or not. To create a classification mask, the human expert visualized the change detection images in RGB, where R[VV]\text{reference}, G[VV]\text{activity}, B[VV]\text{reference}. This visual classification is considered the golden standard and we, therefore, use it as ground truth to train and evaluate our model. The whole dataset contains a total of 6345 avalanches, 3,667,355,474 pixels are classified as “non-avalanche” and 712,945 (0.000194 of the total) as “avalanche”.

For each scene, we constructed three data channels using the difference of the horizontal and vertical polarization between the reference and the activity image: VV = VV\text{activity} - VV\text{reference}, VH = VH\text{activity} - VH\text{reference}. Then, we scaled the channels values to [0,1] (see Fig. 1(a,b)). We also included the point-wise product of the difference images squared as an additional channel: VVVH = VV^2 * VH^2 (see Fig. 1(c)). We did not consider radar shadow, layover masks, or land masks depicting avalanche runout zones. Especially the latter was deliberately excluded since it is not available for all areas.

A. Additional topographical information

To include topographical information, indicative of avalanche terrain, we generated two feature maps for each S1 scene obtained from the digital elevation model (DEM), that is freely available for entire Norway in 10m pixel resolution.

Slope angle: The slope angle feature map is directly computed by taking the slope gradient of the DEM (see Fig. 1(d)). Comparing the slope angle distribution of avalanche and non-avalanche pixels results in two distinctively different distributions. In particular, avalanche pixels are mostly concentrated around [20, 35] degrees (see Fig. 2). We thus concluded that the slope angle can be exploited to discriminate between the two classes.

Potential angle of reach (PAR): The angle of reach of an avalanche, often denoted \( \alpha \), is the angle between the horizontal and the line intersecting the highest topographical point where an avalanche releases and the point of furthest avalanche runout \[13\]. Unfortunately, we did not have information about the highest topographical point available, as it is uncertain if the release zone part of an avalanche is always detectable in S1 scenes. We therefore introduce the potential angle of reach
(PAR), denoted $\tilde{\alpha}$ as illustrated in Fig. [3]. Considering the angle found between the horizontal and the line intersecting the point of interest and a point in the neighboring release zones, the PAR is obtained by maximizing this angle for all points in the neighboring release zones. Defining release zones as regions with a slope of 30-50 degrees and limiting the distance to the release zone points to 4 kilometres, we obtain the PAR depicted in Fig. [1(e)]. Fig. [3] depicts the distribution of the PAR for avalanche and non-avalanche pixels. It is possible to see that for avalanche pixels the distribution is more regular and has a single peak centred around 40 degrees. We thus concluded that the PAR is informative since the two distributions are different for the two classes.

$$\tilde{\alpha} = \max_x \theta(x)$$

Fig. 3. Definition of the potential angle of reach $\tilde{\alpha}$, where $\theta(x)$ denotes the angle between the horizontal and the line drawn from a point in a release zone, denoted $x$, to the point of interest.

Fig. 4. Distribution of the PAR for avalanche and non-avalanche pixels

### III. Method

The FCN network used for segmentation is a U-Net architecture [14], which is suitable when a low to medium amount of training data is available. The network consists of an encoder and a decoder part, respectively depicted in blue and red in Fig. [5]. The encoder hierarchically extracts feature maps that indicate the presence of patterns in the image. By reducing the spatial dimensions and increasing the number of filters, the deeper layers capture patterns of increasing complexity and larger spatial extent in the input image. The decoder gradually transforms the high-level features and, in the end, maps them into the output. The output is a binary segmentation mask, which has the same height/width of the input and indicates which are the pixels that belong to the avalanche class. The skip connections link the feature maps from the encoding to the decoding layers, such that some information can bypass the bottleneck located at the bottom of the “U” shape. In this way, the network still learns to generalize from the high-level latent representation but also recovers from the intermediate representations of the spatial information through a pixel-wise semantic alignment.

Fig. [5] shows the architecture details: the number in each Enc/Dec Block indicates the number $n$ of 3x3 filters in the Conv($n$) layers. The encoder reduces the spatial dimension with max pooling, while the decoder restores it through bilinear upsampling. Each block contains 2 Batch Norm [15] and one Dropout layer [16], which are respectively used to facilitate the training convergence and improve the model generalization. The last encoder block (Enc Block 512 in Fig. [5]) does not have Dropout, while the last decoder block (Dec Block 32) is followed by a Conv layer with one 1x1 filter and a sigmoid activation. Since the network is fully convolutional (there are no dense layers), it can process images of variable size.

#### A. Class balance

Avalanches are small objects and the avalanche class is highly under-represented in the dataset (avalanche pixels are 0.019% of the total). Therefore, a trivial model that classifies each pixel as “non-avalanche” would reach a classification accuracy of 99.98%. A solution to handle class unbalance is to differently weight the loss relative to the pixels of the different classes, so that the network is more penalized when it misclassifies the underrepresented class [9]. Specifically, we configured the loss...
to give twice more importance to the classification errors on the avalanche pixels. We also experimented with loss functions specifically designed to handle class unbalance, such as the Jaccard-distance loss \cite{17} and the Lovász-Softmax loss \cite{18}, but we obtained worse results than optimizing the FCN using binary cross-entropy loss with class balancing.

B. Data augmentation

To avoid overfitting during training and enhance the model generalization to new data, we perform data augmentation by randomly applying (on the fly) horizontal and vertical flips, horizontal and vertical shifts, rotations, zooming and shearing to the training images. To ensure consistency, the same transformations on the input are also applied to the ground truth avalanche masks.

To limit border effects and produce smoother and more accurate predictions, we perform test time augmentation. First, we sample from the whole scene overlapping windows, so that they have a 50% overlap with the left, right, bottom and upper neighboring windows. Then, we compute predictions for all the possible 90° rotations and flips of the windows, which is useful to disrupt orientation-specific biases learned by the network. Finally, the resulting predictions are combined using a 2nd order spline interpolation, which further reduces the variance in the predictions.

C. Attention mask

Following our hypothesis that the PAR feature map can highlight areas where it is more highly likely find an avalanche, we propose a neural attention mechanism \cite{19} that generates an attention mask conditioned on the PAR. The mask encourages the segmentation procedure to focus on specific regions of the input. Specifically, we use a small network that takes as input the PAR and generates the attention mask that is, in turn, applied element-wise to the pixels of the SAR channels VV and VH before they are fed into the segmentation network (see Fig. \ref{fig:network}). The attention network consists of three stacked Conv layers with 32 3x3 filters and ReLU activations and a Conv layer with 1 3x3 filter and sigmoid activation. The attention network has a small receptive field (7 pixels), meaning that each attention value only depends on the local PAR. This is acceptable, since the PAR yields highly non-localized features from the DEM and captures long-range relationships in the scene.

The attention network is also fully convolutional and is jointly trained with the rest of the segmentation architecture. Our solution allows to directly learn from the data how to generate and how to apply the mask, in a way that is optimal for the downstream segmentation task. This is a more flexible approach than masking out parts of the input (e.g. by applying pre-computed runout masks), or directly pre-multiplying the SAR channels with the PAR feature map.

D. Network training

We trained the FCN by feeding it with small square patches, rather than processing entire scenes at once, which would also be unfeasible due to the memory limitations of the graphic card\footnote{Two Nvidia GTX2080 were used to train and evaluate the network}. By using small patches it is possible to inject stochasticity in the learning phase by randomly shuffling and augmenting the data at each epoch. This prevents overfitting and decreases the chances of getting stuck in local minima. We experimented with patches of 160 × 160 or 256 × 256 pixels, which is a size compatible with the receptive field of the filters in the innermost network layer (Enc Block 512), which is 140. After preliminary experimentation, we obtained the best performance with the 160 × 160 patches.

Out of the 118 available S1 scenes, one scene with date 17-Apr-2018, which contains 99 avalanches, was isolated from the rest and used for testing. To build the training set we considered only the patches containing at least 1 pixel classified as “avalanche” by the human expert. We ended up with ≈ 35,000 patches, of which 10% were used as validation set for model selection and evaluation. The network is trained with Adam optimizer \cite{20} with default parameters, using mini-batches of size 16 and dropout rate 0.4.

IV. RESULTS AND DISCUSSION

Examples of FCN predictions are depicted in Fig. \ref{fig:predictions}. Since the networks predict continuous values in [0,1], the predictions are binarized by thresholding them at 0.5.

Since the avalanche class is highly under-represented, accuracy is not a good measure to quantify the performance and, therefore, we evaluated the quality of the segmentation result by using different metrics. The first is the F1 score, which is computed at the pixel level and is defined as

\[ F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \]

where precision is defined as \( \frac{TP}{TP + FP} \) and recall is \( \frac{TP}{TP + FN} \) (TP = True Positives, FP = False Negatives, FN = False Negatives). The F1 score is also computed during training on the validation set for early stopping and for saving the best model. To evaluate the segmentation results at a coarser resolution level, we considered the bounding boxes for each avalanche in the ground truth mask and in the predicted one. TP are measured as the number of bounding boxes in the ground truth that have a non-zero intersection with a bounding box in the prediction. In a similar way, we compute the FP and FN. To quantify how much the bounding boxes in the ground truth and in the prediction overlap, we computed the intersection over union (IoU):

\[ \text{IoU} = \frac{\text{Area of bounding boxes intersection}}{\text{Area of bounding boxes union}}. \]

We compared the proposed method with the algorithm currently used in our production pipeline for automatic avalanche detection \cite{2}. This state-of-the-art algorithm represents our baseline and it is primarily driven by change detection and filtering methods to enhance potential avalanche features. Dynamic thresholding based on the statistics of image pairs controls the final delineated features. The method is to a large extent dependent on additional input layers such as slope,
Fig. 6. For each patch, the Attention Net generates an attention mask from the PAR features and applies it to the VV and VH SAR channels. Then, the masked SAR channels and the slope (not masked), are fed into the U-Net that is jointly trained to segment the patch. The VVVH channel is not shown for conciseness.

Fig. 7. Examples of prediction on individual patches. From the left: i) VVVH input channel fed to the FCN; ii) ground truth labels manually annotated by the expert; iii) raw output of the FCN; iv) FCN output thresholded at 0.5. Note that those are the row outputs of the FCN and no test time augmentation is applied at this point.

### Table I

| Method | F1 (%) | IoU (%) | TP (#) | FN (#) | FP (#) |
|--------|--------|---------|--------|--------|--------|
| Baseline | 38.13  | 33.11  | 44     | 45     | 11     |
| FCN    | 66.6   | 54.3    | 72     | 17     | 32     |

**Segmentation results from the test image with 99 avalanches.** We report the F1 score (in percent), intersection over union of the bounding boxes (in percent), true positive (correct hits), false negative (missed avalanches detection), and false positive (false avalanches detection).

vegetation maps and runout zone information in order to restrict the areas where features are allowed to be detected, thereby reducing the risk of false alarms as much as possible.

Tab. I reports the results obtained on the test image. Compared to the baseline, FCN achieved a much higher agreement with the manual labels, as indicated by the higher F1 and IoU values. Out of the 99 avalanches in the test image, FCN correctly identified 72 of them and missed 17. However, most of the FN are small avalanches and difficult to detect. FCN also identified 32 FP and most of them are due to particular terrain structures, which cause high backscatter that resemble avalanches Fig. 8. Interestingly, some of those FP are actual avalanches that have been overlooked during the manual annotation. More examples of the FCN’s performance are depicted in Fig. 8.

### Ablation study

The ablation study consists in removing some “feature” of the model or of the data and seeing how that affects performance. In particular, we study how much each SAR channel and the topographical feature maps contribute to the segmentation. We also evaluate the difference in concatenating the PAR to the other input channels or using it to compute the attention mask, as described in Sect. III-C. The results reported in Tab. II indicate that the most important improvement comes from including the difference in the VH channel, compared to using the VV channel alone. By adding the slope and PAR features it is possible to further increase the segmentation performance. Finally, the proposed attention mechanism allows to better exploit the information yield by the PAR feature, compared to just concatenating it to the other channels.

### V. Conclusions

In this work, we propose the first deep learning approach for saliency segmentation of avalanches in Sentinel-1 SAR images. As input channels to the network, we computed feature maps using radar backscatter information and its temporal difference as well as topographical information (slope angle and potential...
angle of reach). Ground truth to train the network came from manual labeling of avalanche pixels by a human expert. A total of 118 Sentinel-1 SAR images were labeled, of which 117 were used for training and one single image was used for testing and evaluation.

The feature maps were fed along with the SAR images into a Fully Convolutional Network, which was trained to perform avalanche segmentation. The network was extended with an additional block, jointly trained with the rest, which computes an attention mask conditioned on the potential angle of reach. The mask was applied to the input SAR images, to let the FCN's performance on SAR images with different snow conditions (wet or dry).

The results show the effectiveness of the proposed method, improving the F1 score of 38.1% achieved by the baseline algorithm to 66.6% achieved by the FCN, when comparing to the manual labeling of the human expert. Our model only misses some of the smaller avalanches, while detecting additional avalanches that have been missed by the expert. In a next step, we aim to extend our test dataset in order to evaluate the FCN's performance on SAR images with different snow conditions (wet or dry).

**REFERENCES**

[1] M. Eckerstorfer, E. Malnes, and K. Müller, “A complete snow avalanche activity record from a Norwegian forecasting region using Sentinel-1 satellite-radar data,” *Cold Regions Science and Technology*, vol. 144, pp. 39–51, 2017.

[2] H. Vickers, M. Eckerstorfer, E. Malnes, and A. Doulgeris, “Synthetic aperture radar (sar) monitoring of avalanche activity: An automated detection scheme,” *SCIA*, 2017.

[3] 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 Geoscience and Remote Sensing Magazine*, 2017.

[4] Y. Zhou, H. Wang, F. Xu, and Y.-Q. Jin, “Polarimetric sar image classification using deep convolutional neural networks,” *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 12, pp. 1935–1939, 2016.

[5] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, “Urban land cover classification with missing data modalities using deep convolutional neural networks,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 6, pp. 1758–1768, 2018.

[6] O. A. Penatti, K. Nogueira, and J. A. Dos Santos, “Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?” in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 44–51.

[7] L. T. Luppino, F. M. Bianchi, G. Moser, and S. N. Anfinsen, “Unsupervised image regression for heterogeneous change detection,” *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–16, 2019.

[8] ———, “Remote sensing image regression for heterogeneous change detection,” in *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2018, pp. 1–6.

[9] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, “Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks,” in *CVPR*, 2016.

[10] P. E. Kummervold, E. Malnes, M. Eckerstorfer, I. Arntzen, and F. M. Bianchi, “Avalanche detection in sentinel-1 radar images using convolutional neural networks,” in *International Snow Science Workshop*, 2018.

[11] S. Sinha, S. Giffard-Roisin, F. Karbou, M. Deschatres, A. Karas, N. Eckert, C. Cofléou, and C. Monteleoni, “Can avalanche deposits be effectively detected by deep learning on sentinel-1 satellite sar images?” 2019.

[12] A. U. Waldeland, J. H. Reksten, and A.-B. Salberg, “Avalanche detection in sar images using deep learning,” in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 2386–2389.

[13] S. Bakkehui, U. Domaas, and K. Lied, “Calculation of snow avalanche runout distance,” *Annals of Glaciology*, vol. 4, p. 24–29, 1983.

[14] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *MICCAI*, 2015.

[15] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167, 2015.

[16] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting.” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[17] G. Csurka, D. Larlus, F. Perronnin, and F. Meylan, “What is a good evaluation measure for semantic segmentation?,” in *BMVC*, vol. 27. Citeseer, 2013, p. 2013.

[18] M. Berman, A. Ramen Troli, and M. B. Blaschko, “The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4413–4421.

[19] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *International conference on machine learning*, 2015, pp. 2048–2057.

[20] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations (ICLR)*, 2015.