SEEING THE BIGGER PICTURE: ENABLING LARGE CONTEXT WINDOWS IN NEURAL NETWORKS BY COMBINING MULTIPLE ZOOM LEVELS

Konrad Heidler [1,2], *Lichao Mou* [1,2], *Xiao Xiang Zhu* [1,2]

[1] Data Science in Earth Observation, Technical University of Munich (TUM), Munich, Germany
[2] Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Wessling, Germany

ABSTRACT

When adopting deep learning methods for remote sensing applications, the data usually needs to be cut into patches due to hardware limitations. Clearly, this practice discards a lot of contextual information as the model's information is limited to imagery from the given patch. We propose a memory-efficient way around this limitation by using multiple patches of varying spatial extents on different resolution levels. Finally, this new approach is evaluated for the task of automated sea ice charting, where the added contextual information is shown to be beneficial to model performance.

Index Terms**—** Image segmentation, Multiresolution, Synthetic aperture radar, Sea ice

1. INTRODUCTION

The deep learning methods used in remote sensing are often based on computer vision methods which have been developed for other types of imagery, like front-view photography [1]. This means that these methods are based on assumptions that may not hold for remote sensing imagery, resulting in discrepancies between the approaches and the actual data.

One such difference we would like to highlight here, is the completeness of information within an image. In photography-based data, like ImageNet [2], the images are largely self-contained. All the information that is needed to correctly classify or segment the image is usually contained within the image's boundaries. In remote sensing, the situation is different. Here, the common practice is to crop patches from larger satellite or aerial acquisitions. This means that the image boundaries are only arbitrary cuts made to fulfill memory and processing requirements, and important information might be cut off by this practice (cf. fig. 1). We set out to find a mitigation to this problem for the task of semantic segmentation, that is to provide a deep learning approach which can take into account neighboring data on a larger scale. Due to the aforementioned memory limits, this solution needs to be more sophisticated than simply increasing the patch size,

A key ingredient in many current state-of-the-art computer vision models is the concept of processing data on multiple scales. Following the introduction of pooling layers, this idea has led to successful approaches like encoder-decoder architectures [3], or feature pyramid networks [4]. As they are still based on the paradigm of "the image is all information we have", their lower-resolution feature maps are of accordingly smaller size than the input image. This is the point of attack for a new framework tailored to the remote sensing setting. As there is more data available beyond the limits of the original image, the lower resolution feature maps can be extended to a larger size, to take into account more information from the surroundings.

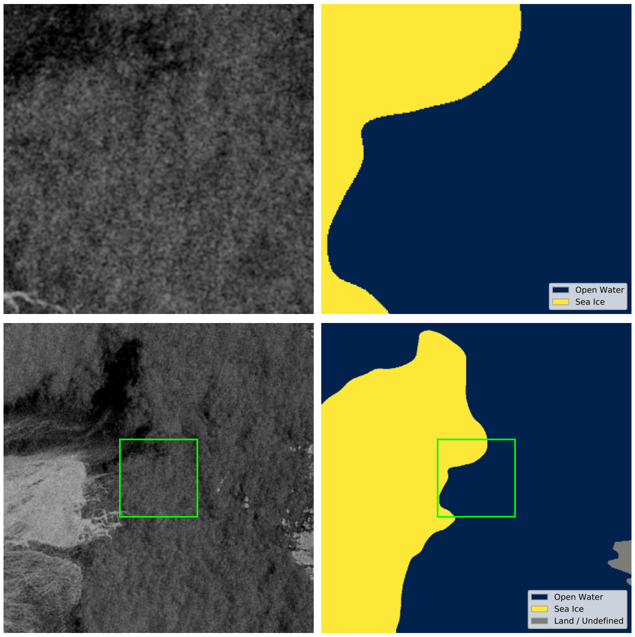

Fig. 1. Top: Sentinel-1 imagery (HH polarization) and corresponding sea ice chart at full resolution. Bottom: The same location (green square) zoomed out by a factor of 4. Zooming out helps seeing the bigger patterns that are necessary for the correct classification.

When providing downsampled versions of the surroundings of a tile as additional inputs to the lower-resolution layers, the model has a way of using this contextual information without increasing the memory footprint much. In a way, this

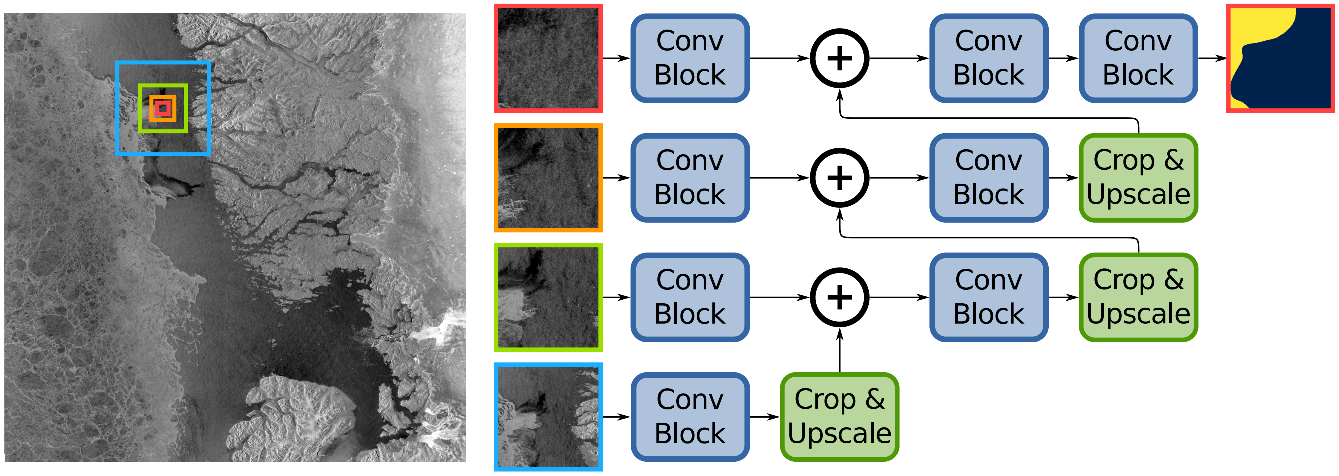

Fig. 2. Proposed network architecture. Centered at the middle of the selected tile, additional crops at ½, ¼ and ⅛ times the full resolution are taken as additional inputs. All input maps have the same size of 256×256 pixels, however from the bottom to the top, the spatial resolution increases by a factor of two for each row.

approach mimics how humans usually look at aerial or satellite images. After getting a good overview at a lower zoom level, one can then zoom in and look at an area in full resolution.

2. METHOD

The practice of tiling large aerial or satellite images into smaller patches effectively prevents the model from looking beyond the limits of that single tile. Clearly, this imposes a limit on the spatial context that the neural network can ingest. This restriction is especially severe for segmentation approaches, where pixels close to the edge are missing nearly half of their spatial context.

We propose a mitigation for this loss of context by "zooming out". By that, we mean that the surrounding data is not completely discarded, but instead processed at a lower resolution. Following the basic ideas behind feature pyramid networks [4], our framework performs processing steps at a number of different resolution levels. However, our proposed architecture bases its decisions on a larger spatial context by not only taking a single image as its input, but multiple ones. In addition to the full-resolution tile, these other inputs are given at 2^{-n} times the full resolution. This practice makes them coincide with the spatial resolution of feature maps that have been down-sampled n times.

Intuitively speaking, this practice reduces the spatial resolution perceived by the model as it looks farther away from the scene of interest instead of completely blocking the surroundings from view. Thanks to the iteratively reduced resolution, the model can judge a scene based on a comparatively large spatial context without exploding memory consumption.

The information derived from the coarser input patches is merged with the features at higher resolutions in an iterative fashion. After initializing feature maps for each scale, the feature maps are merged from the bottom up. First, the very coarse features are merged, so that an informative representation of the contextual surroundings is aggregated. Then, these coarse features are merged into successively finer feature maps, until the full image resolution is reached. At the final stage of the pipeline, a prediction is calculated for the smallest, highest-resolution patch.

2.1. Data Preprocessing

The data preprocessing pipeline is an integral component of our approach. In addition to cropping the patches from the full-resolution tile, the lower resolution input patches need to be created as well. This can be efficiently done by successively creating downsampled versions of the full tile. For each downsampling step, neighborhoods of 2×2 pixels are averaged, resulting in an image of half the previous resolution. This process is repeated three times so that a pyramid of images at 4 resolution levels is obtained. Afterwards, it is straightforward to cut multiresolution tiles from this pyramid by cropping patches of the same size, but with the strides halving at each resolution level. Tiles close to the border need special treatment, as the zoomed-out input patches can extend beyond the original tile's dimensions. For these cases, reflection padding is used.

2.2. Merging feature maps

On each resolution level, feature maps are initialized with a convolutional block. After this initialization step, the feature maps need to be merged into a unified representation that

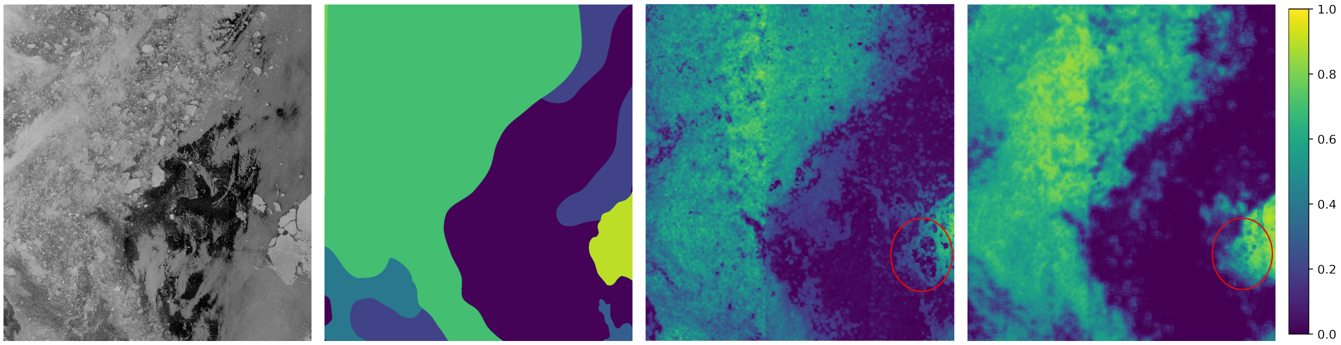

Fig. 3. Example for model predictions after training on soft labels ("Prob."). From left to right: Input imagery (HH channel), ground truth, ASPP-CNN prediction (image taken from [5]), prediction from our model. Marked in red: a region that is classified more accurately with the enlarged context window.

combines contextual and local information to provide a basis for an accurate segmentation. We propose to merge the features from the bottom up, in a step-by-step fashion, as outlined in Fig. 2.

In order to merge corresponding features together, great care needs to be taken when joining the branches from different resolution levels. The "Crop & Upscale" block achieves this by first cropping the feature map to its central region of half the extents. Then, the feature map is bilinearly upsampled by a factor of two. The resulting upsampled feature map is thus spatially aligned with the feature maps at the next higher resolution level. Merging is done by element-wise addition of the feature maps, a method which is known to provide good gradients at training time [6]. It may seem that this simple merging procedure could lead to loss of detail, as the lower resolution branches appear to outweigh the high resolution features in this architecture. However, it should be noted that the magnitude of the features before the merging is a learned quantity, so the network can figure out the optimal mixing behavior to ensure that enough detail is preserved. Our experiments further confirm this interpretation.

2.3. Final predictions

After merging the features from all resolution levels, the final feature map has full resolution, but is enriched with contextual information from the coarser levels. In terms of current computer vision approaches, the network up to the last merging step can be viewed as a remote-sensing specific backbone network, that leverages a multiresolution pyramid to calculate its feature representation. The aggregated information from this multiresolution backbone is then used to predict the final segmentation map with a segmentation head.

After this rich contextual aggregation in the backbone, calculating the final segmentation is fairly easy for the network. Our experiments show that it is sufficient to use a comparably shallow prediction head that consists of only two convolutional blocks.

3. EXPERIMENTS & RESULTS

To evaluate the improvements made in our approach, we train and validate our model on the *AI4Arctic ASIP Sea Ice Dataset*[1], which consists of Sentinel-1 scenes as well as lower resolution AMSR2 microwave radiometry data of the waters around Greenland, and corresponding sea ice annotations.

The results are compared with the ones obtained by [5], who put together the dataset and trained a CNN on the data. As the task of sea ice charting does require quite a bit of contextual information, the authors employ atrous spatial pyramid pooling (ASPP) [7].

ASPP is conceptually quite similar to our approach, as it combines information at multiple scales using dilated convolutions. However, it is still bound to the information present within each tile and cannot look beyond its borders. This makes it a very interesting competitor to our method, both approaches work on multiple scales, but ours has an improved spatial context. Therefore we can directly attribute any observed performance gains to the enlarged context window.

To fully make use of the available data in the dataset, we include the low resolution AMSR2 microwave radiometry data by resampling it to $^1/_8$ of the full resolution. This resampled version can then be easily concatenated to the coarsest input patch along the channel dimension.

We adopt the validation protocol of creating the validation split by withholding 10% of the scenes from training.

The dataset is labeled not just with binary annotations, but instead with the sea ice percentage, rounded to the closest multiple of 5%. So for training a classifier, there are the following two options [5]:

1. Directly using the percentage values as *soft labels* in the cross-entropy loss formula
2. Thresholding at a certain percentage (here: 10%) to obtain *hard binary labels*

[1] Available at https://data.dtu.dk/articles/dataset/ASIP_Sea_Ice_Dataset_-_version_1/11920416

Model	Accuracy (Prob.)	Accuracy (Thresh.)
ASPP-CNN [5]	94.26%	95.29%
ours	96.59%	96.46%

Table 1. Validation results on the ASIP Sea Ice Dataset.

Both methods have their theoretical advantages, the first one trains the model to predict the probability of ice being present for each given pixel. However, the "hard" labels in the second approach encourage a sharper delineation of the edges in the ice chart. Therefore, both of these approaches are evaluated, and denoted by "Prob." and "Thresh.", respectively.

Table 1 quantifies the performance of the trained models compared to the results obtained in [5]. It can be seen that the straightforward implementation of enlarged spatial context windows through multiple zoom levels can indeed improve the segmentation results compared to conventional patch-based CNN approaches.

4. CONCLUSION

We have presented a new approach for applying deep learning methods to remote sensing data that, other than existing methods, allows the model to look beyond the limits of the current tile. This is achieved by cropping patches not only at the full resolution but at lower resolutions as well. The smaller resolution allows these additional patches to cover a larger area. While this approach requires some changes to the data input pipeline, we are convinced that some segmentation tasks can greatly benefit from this approach. As an example, we have showed that our method improves the results for automated sea ice charting over existing methods.

As next steps, we would like to apply the ideas behind this method to other segmentation tasks and datasets. Finally, an adaptation of this framework to other tasks like scene classification or object detection is easily possible by changing the architecture after the last merging step, allowing the enriched context representations to be leveraged for these tasks. It remains interesting to check how much the contextual information will help on higher resolution datasets or more complex classification tasks.

5. REFERENCES

[1] Xiao Xiang Zhu, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[2] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 248–255.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, Eds., Oct. 2015, pp. 234–241.

[4] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature Pyramid Networks for Object Detection," *arXiv:1612.03144 [cs]*, Apr. 2017.

[5] David Malmgren-Hansen, Leif Toudal Pedersen, Allan Aasbjerg Nielsen, Matilde Brandt Kreiner, Roberto Saldo, Henning Skriver, John Lavelle, Jørgen Buus-Hinkler, and Klaus Harnvig Krane, "A Convolutional Neural Network Architecture for Sentinel-1 and AMSR2 Data Fusion," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–13, 2020.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proc. 2016 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2016, pp. 770–778.

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *arXiv:1606.00915 [cs]*, May 2017.