Use of Commercial Satellite Imagery to Monitor Changing Arctic Polygonal Tundra

Amit Hasan, Mahendra Udawalpola, Anna Liljedahl, and Chandi Witharana

Abstract

Commercial satellite sensors offer the luxury of mapping of individual permafrost features and their change over time. Deep learning convolutional neural nets (CNNs) demonstrate a remarkable success in automated image analysis. Inferential strengths of CNN models are driven primarily by the quality and volume of hand-labeled training samples. Production of hand-annotated samples is a daunting task. This is particularly true for regional-scale mapping applications, such as permafrost feature detection across the Arctic. Image augmentation is a strategic "data-space" solution to synthetically inflate the size and quality of training samples by transforming the color space or geometric shape or by injecting noise. In this study, we systematically investigate the effectiveness of a spectrum of augmentation methods when applied to CNN algorithms to recognize ice-wedge polygons from commercial satellite imagery. Our findings suggest that a list of augmentation methods (such as hue, saturation, and salt and pepper noise) can increase the model performance.

Introduction

A network of polygonal patterns appears in the tundra due to the cracking and subsequent development of ice wedges. Ice-wedge polygons (IWPs) are one of the most common landforms across the Arctic tundra lowlands. Early studies (Leffingwell 1919) described two major types of IWPs: (1) polygons with elevated blocks or high-centered polygons and (2) polygons with depressed blocks or low-centered polygons. The microtopography associated with IWP controls a multitude of functions of the Arctic ecosystem (Kutzbach *et al.* 2004), such as permafrost and hydrologic dynamics from local to regional scales, due to the linkages between microtopography and the flow and storage of water (Liljedahl *et al.* 2016), vegetation succession (Magnússon *et al.* 2020), and permafrost dynamics (Lara *et al.* 2020). Widespread ice-wedge degradation is transforming low-centered polygons into high-centered polygons in a rapid phase (Steedman *et al.* 2016).

The entire Arctic has been imaged by high-spatial-resolution commercial satellite sensors, producing sheer volumes of data. Imagery archives are quickly morphing to petabyte scale. While studies have been conducted on vegetation dynamics (Verdonen *et al.* 2020), phenology (Zheng *et al.* 2020), vegetation classification (Davidson *et al.* 2016), and spectral and seasonal variation of leaf area index (Juutinen *et al.* 2017), imagery-derived products lag behind. We are in the process of translating these big imagery resources to Arctic science–ready products. Our ongoing research investigates the automated detection of IWPs from commercial satellite imagery.

The successful implementation of deep learning (DL) convolutional neural nets (CNNs) in computer vision applications has received a great deal of interest from the remote sensing community (Ma *et al.* 2019). There has been an upsurge of recent research that exhibits DLCNN applications in a multitude of remote sensing classification problems, such as land use and land cover types of detection (Paoletti *et al.* 2019; Zhang *et al.* 2019), agricultural crop mapping (Zhong *et al.* 2019), feature extraction from remote sensing images (Romero *et al.* 2016), object localization (Long *et al.* 2017), cloud detection (Xie *et al.* 2017), and disaster recognition (Liu & Wu 2016). DLCNNs perform well in terms of object detection (Zhao *et al.* 2019), image segmentation (Rizwan I Haque and Neubert 2020), and semantic object instance segmentation (Lateef and Ruichek 2019). An array of DLCNN architectures have been developed, trained, and tested with different types of imagery. Each of these architectures has its own advantages and disadvantages with respect to computation time and resources. Among many others, Mask R-CNN, U-Net, and Deeplab V3+ stand out as superior methods in semantic object instance segmentation. Researchers used Deeplab V3+ with the Pascal VOC data set and achieved 89% intersection over union (IoU). In a separate biological image segmentation data set, the U-Net model achieved a total of 85.5% IoU (Karimov *et al.* 2019). There is an increasing interest in the application of the Mask R-CNN model for Earth science applications (Su *et al.* 2019; Bhuiyan *et al.* 2020; Carvalho *et al.* 2021; Mahmoud *et al.* 2020; Zabawa *et al.* 2020; Zuo *et al.* 2020). Previous studies have shown promising results found by the implementation of DLCNN with commercial satellite imagery (Zhang *et al.* 2018; Bhuiyan *et al.* 2020; Witharana *et al.* 2020). By design, inferential strengths of CNN models are fueled largely by the quality and volume of hand-labeled training data. Production of hand-annotated samples is a daunting task. This is particularly true for regional-scale mapping applications, such as permafrost feature detection across the Arctic, where landscape complexity would spontaneously inflate the semantic complexity of submeter-resolution imagery. Additionally, image dimensions, multispectral channels, imaging conditions, and seasonality, coupled with multi-scale organization of geo-objects, pose extra challenges on the generalizability of DLCNN models. Image augmentation is a strategic "data-space" solution to synthetically inflate the size and quality of training samples without additional investments on hand annotations. A plethora of augmentation methods have been proposed under the auspices of two general categories: data warping and oversampling (Shorten and Khoshgoftaar 2019). The performance of image augmentation methods depends largely on the image recognition problem on hand and the characteristics of the underlying data. Researchers have used color augmentation techniques for skin lesion segmentation and classification (Galdran *et al.* 2017), geometric transformation with chest X-ray for the screening of COVID-19 (Elgendi *et al.* 2021), and noise injection techniques for plant leaf disease detection (Arun Pandian *et al.* 2019).

In this study, we have investigated the efficacy of 17 augmentation methods in relation to IWP detection. We relied on the Mask R-CNN algorithm as the base model in the training and the prediction of IWPs. The Mask R-CNN model itself has a lot of room to modify and tweak the default parameters (He *et al.* 2017). The backbone of the model is a convolutional neural network. This can be changed to different types of CNN models; we used the ResNet-50 structure (He *et al.* 2015) as the backbone. To initialize the model, we have practiced the transfer learning approach. In this approach, the model is already trained based on another hand-labeled data set. Our backbone was pretrained based on

Amit Hasan, Mahendra Udawalpola, and Chandi Witharana are with the University of Connecticut, Storrs, CT 06269 (amit.hasan@uconn.edu).

Anna Liljedahl is with the Woodwell Climate Research Center, Falmouth, MA 02540.

Contributed by Alper Yilmaz, August 30, 2021 (sent for review August 30, 2021).

Photogrammetric Engineering & Remote Sensing
Vol. 88, No. 4, April 2022, pp. 255–262.
0099-1112/22/255–262

doi: 10.14358/PERS.21-00061R2

the ImageNet data set. We retrained the Mask R-CNN model with different augmentation methods using our data set so that the model could be used for the detection and segmentation of the IWPs. One of the major weaknesses of the DLCNNs is that spectral and spatial variations in the training data set affect the model performance (Grm *et al.* 2018). In our data, the spectral, spatial, and textural characteristics of IWPs vary based on the tundra vegetation types (Stow *et al.* 1993; Sturtevant *et al.* 2013; Mikola *et al.* 2018). Thus, separate Mask R-CNN models are trained for different tundra types.

The main goal of this study is to explore the potential of augmentation methods on top of a state-of-the-art DLCNN method (Mask R-CNN) to characterize the tundra IWP landscape as well as to assess the change in the model performance when trained with separate tundra types. We conducted a multi-step quantitative assessment to assess the precision, recall, F1 score, and overall accuracy of the prediction results from each of the augmentation scenarios.

Methods

Study Area

We extracted a total of 696 image tiles of varying dimensions (such as 292 × 292, 345 × 345, 507 × 507, and 199 × 199 pixels) out of seven satellite imagery scenes from the Norths Slope of Alaska, Prince Patrick Island, Banks Island, Inuvik in Canada, and Nizhnekolymskiy Ulus in Russia (Figure 1). These areas are covered mostly by tussock and non–tussock sedge tundra, sedge/grass, moss wetland, and other types of tundra. We hand annotated a total of 25,509 polygons (15,989 low-centered and 9520 high-centered polygons) from the satellite image scenes.

We prepared three sets of images out of all annotated patches as the training (487 images), test (106 images), and validation (103 images) data sets. The training data set was used for training the model, and the validation data set was used to check model performance while training the model. The test data set was used to calculate the performance of the trained model.

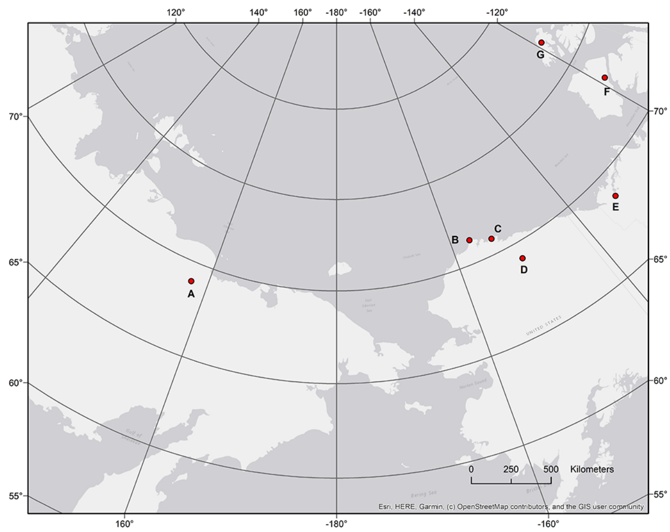


Figure 1. Geographical distribution of study sites: (A) Nizhnekolymskiy Ulus, Russia. (B) Barrow, Alaska, USA. (C) Atqasuk, Alaska, USA. (D) Prudhoe Bay, Alaska, USA. € Inuvik, Canada. (F) Banks Island, Canada. (G) Prince Patrick Island, Canada.

Model Architecture

Our experimental design was centered on the Mask R-CNN model architecture (He *et al.* 2017) (Figure 2). This model is specialized in object detection as well as instance segmentation at the same time.

We built the work flow based on an open-source package, built on Keras and TensorFlow developed by the Mask R-CNN team, that is available on Github (Waleed Abdulla 2017). The Mask R-CNN model consists of a CNN backbone, a region proposal network, and neural networks for predicting classes, bounding boxes, and masks (Figure 2). We used ResNet-50 (He *et al.* 2016) pretrained with the ImageNet data set as the backbone of the Mask R-CNN network. The final outputs of the model consist of the polygons detected inside the bounding boxes as well as in the form of masks and the class names (high-centered or low-centered polygons) corresponding to each of those detected polygons.

Augmentation Methods

Image augmentation is a process that modifies training images in a variety of ways and acts like additional training images to the model. Image augmentation, thus, can boost the performance of DL models by introducing additional training data. In the Mask R-CNN model, it is possible to implement augmentation methods. Table 1 exhibits the augmentation methods that we used in our study.

Some augmentation methods (e.g., flipping) do not change the spectral distribution of the input images, whereas other methods (e.g., Gaussian noise) change the spectral distribution of the input images. Also, all the augmentation methods do not essentially improve the model performance, as we will see in the "Results" section.

Other than the single augmented methods, we have implemented combined augmented methods. For example, we have combined the salt and pepper noise and hue augmentation, saturation augmentation, and hue-saturation augmentation methods into a single pipeline and named it spectral augmentation to get the benefits of all the individual augmentation methods. We also used a sequential combination of the salt and pepper noise augmentation and the FlipLR augmentation method. The last augmentation method, named top 7, includes seven augmentation methods that appeared at the top when ranked by their performance. The performance assessment process is discussed in the section "Accuracy Assessment." Figure 3 lists some of the sample images, showing the effects of augmentation methods with respect to the original image.

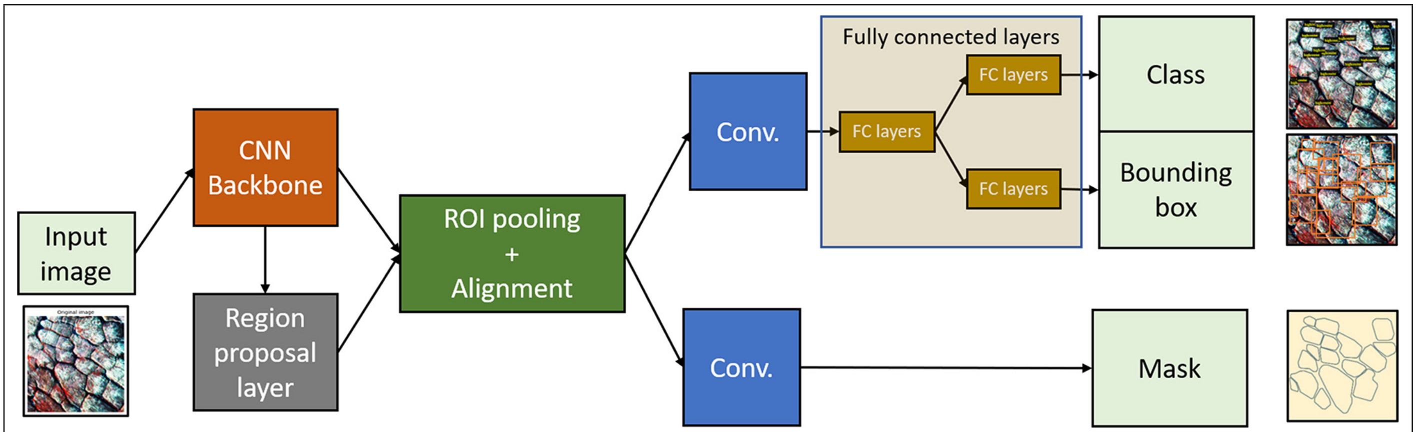


Figure 2. Simplified block diagram of the Mask R-CNN model.

Table 1. Augmentation methods used in this study.

Augmentation Type	Augmentation Methods	Description
Color space transformation	Hue	Multiplies the hue of images by random values
	Saturation	Multiplies the saturation of images by random values
	Hue saturation	Multiplies the hue and saturation of images by random values
	Invert	Subtracts all pixel values from 255
Geometric transformation	Crop	Generates smaller subimages from given full-sized input images
	Flip left to right (FlipLR)	Flips the image horizontally
	Flip up and down (FlipUD)	Flips the image vertically
	Flip left to right and up and down (FlipLRUD)	Combination of FlipLR and FlipUD
	Rotation (x)	Apply affine rotation of x degrees on the y-axis to input data
Noise injection	Gaussian noise	Adds noise sampled from Gaussian distributions
	Salt and pepper noise	Adds salt and pepper noise (noisy white-ish and black-ish pixels) to rectangular areas within the image
Mixed	Salt and pepper and FlipLR	A combination of salt and pepper noise and FlipLR method
	Spectral	A sequential combination of salt and pepper noise, hue, saturation, and hue-saturation augmentation methods
	Top 7 augmentations	A sequential combination of the top 7 augmentation methods based on their mean average precision score on the test data set (Figure 7d), including FlipLR, FlipUD, FlipLRUD, hue saturation, hue, saturation, and salt and pepper noise.

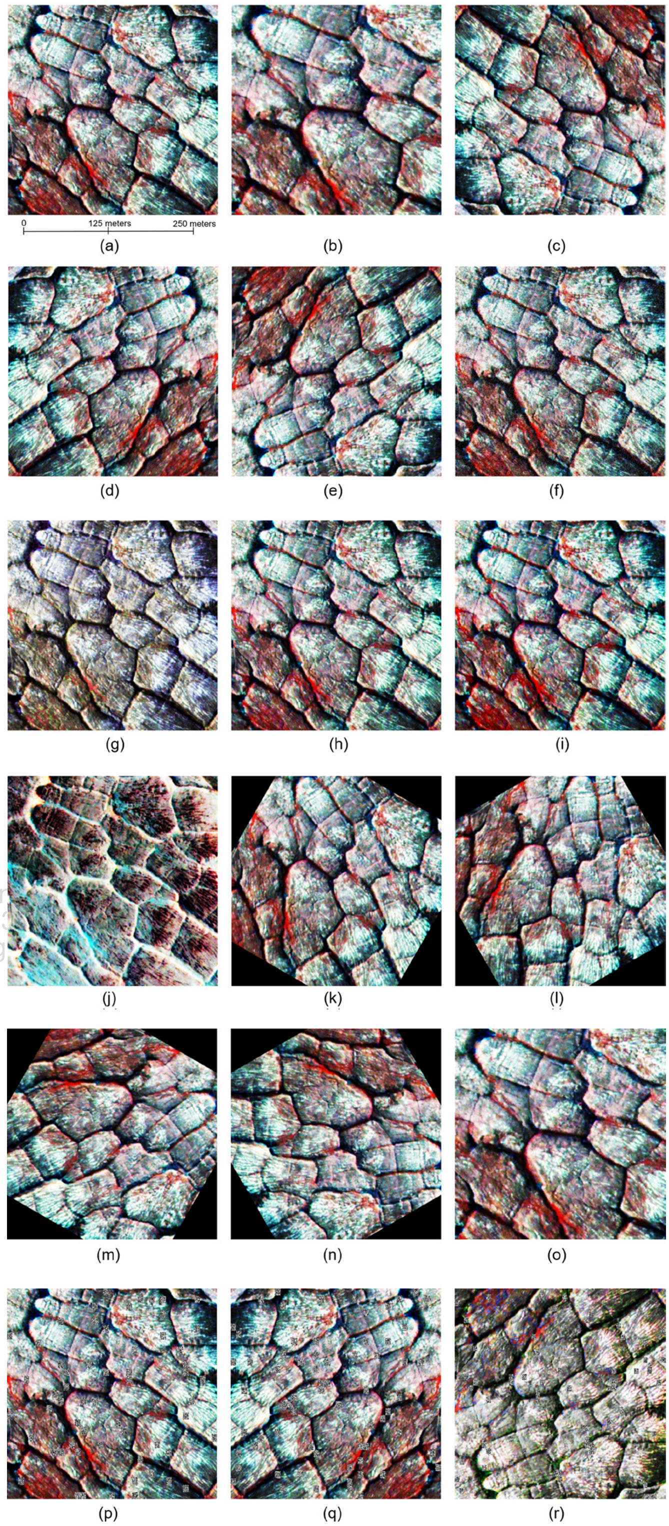


Figure 3. Zoomed-in views of an example original image and corresponding augmented images. (a) Original image. (b) Crop. (c) FlipLRUD (flip left to right and up and down). (d) FlipLR flip left to right). (e) FlipUD (flip up and down). (f) Gaussian noise. (g) Hue saturation. (h) Hue. (i) Saturation. (j) Invert. (k) Rotation (30). (l) Rotation (60). (m) Rotation (120). (n) Rotation (150). (o) Spectral. (p) Salt and pepper noise. (q) Salt and pepper and FlipLR. (r) Top 7 augmentations.

Model Dependency

Different tundra vegetation types exhibit distinct spectral, spatial, textural characteristics, which in turn decide the semantics of overlying IWPs (Liu *et al.* 2017). Landscape complexity translates to the image complexity, affecting DL model performances. Our idea was to implement separate models for separate tundra types and to study the model performance. To achieve this, we selected the best augmentation methods based on their performance and then trained separate models with separate training data sets, each of which will contain only one type of tundra. When the distribution of tundra types in our annotated data is compared to the entire Arctic, we see that our data have different distributions than the original Arctic (Figure 4). However, three of the major tundra types cover more than 70% of our sampled data set. Thus, we have prepared four tundra types named non–tussock sedge (G3) , tussock sedge (G4), sedge/grass (W1), and other tundra types (Others).

Model Training

We used transfer learning approach to retrain the Mask R-CNN model. While doing so, we have taken the ResNet-50 as the CNN backbone of the model. The model was initially trained with the ImageNet data set. The training process was completed in a local machine with an Intel Core i9 CPU with NVIDIA GeForce RTX 2070 SUPER with 8 GB of GPU memory. The training time was not measured, as multiple training processes were run on the local machine at the same time, and based on the GPU load, the training time was varied.

After deciding the augmentation methods and the tundra types, we trained the Mask R-CNN model with minibatches (we changed the step size and batch size based on the memory available in the GPU),

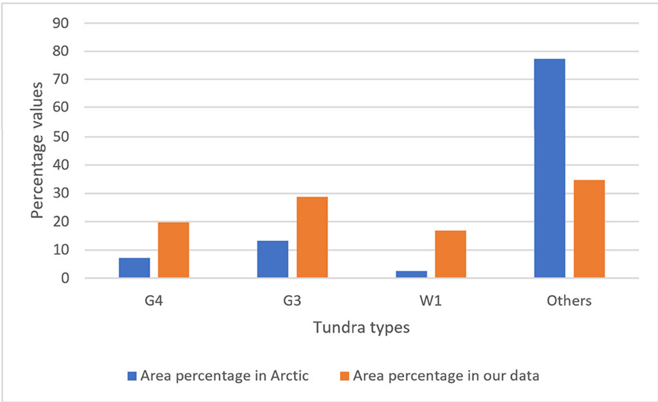


Figure 4. Percent distribution of tundra types, such as tussock sedge (G4), non–tussock sedge (G3), sedge/grass (W1), and other tundra types (Others) on the ground and in the training data. Percent distribution of tundra types on the ground was based on the Circumpolar Arctic Vegetation Map (Raynolds et al. 2019).

learning rate of 0.001, learning momentum of 0.9, and weight decay of 0.0001. We had a total of 487 training image tiles (11,151 low-centered polygons and 6404 high-centered polygons), 103 validation image tiles (2108 low-centered polygons and 1584 high-centered polygons), and 106 test image tiles (2108 low-centered polygons and 1584 high-centered polygons).

To optimize the model, we calculated different losses, such as (1) L1 loss (this defines box regression on object detection systems, which is less sensitive to outliers than other regression loss), (2) Mask R-CNN bounding box loss (this loss indicates the difference between predicted bounding box correction and true bounding box), (3) Mask R-CNN classifier loss (this loss estimates the difference of class labels between prediction and ground truth), (4) mask binary cross-entropy loss (this loss measures the performance of a classification model by observing predicted class and actual class), (5) RPN bounding box loss (this loss identifies the regression loss of bounding boxes only when there is object), and (6) RPN anchor classifier loss (this loss indicates the difference between the predicted RPN and actual closest ground-truth box to the anchor box). The total loss consists of the summation of all these loss values. We prepared the training and validation loss graphs for each of the augmentation methods (Figure 5, see next page) and for each of the tundra types (Figure 6). Based on these graphs, we have selected the best models for each of the augmentation methods or tundra types. In Figure 5, all the models converge at a point, but in Figure 6, the G3 tundra type seems to converge when trained for 200 epochs.

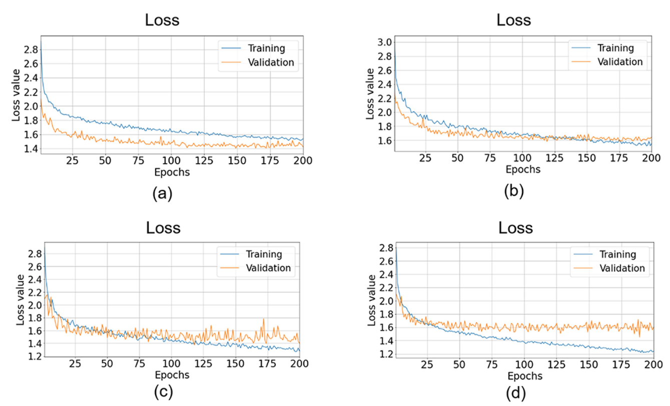


Figure 6. Loss plots for different tundra types: (a) Non–tussock sedge or G3. (b) Tussock sedge or G4. (c) Sedge/grass or W1. (d) Other tundra types.

Accuracy Assessment

We conducted a multistep accuracy assessment for the outputs. The outputs are in the form of class names and binary masks. We calculated the IoU for each of the polygons in the outputs that matched with the polygon classes in the test data set. We set a threshold of the IoU values as 0.5 and considered the polygons above this threshold as correctly classified.

We calculated precision, recall, and F1 score for each of the classes and for each of the images based on Equations 1–3:

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \tag{1}$$

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{2}$$

$$F1 \text{ score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{3}$$

We then calculated the average precision, recall, and F1 score for low-centered and high-centered polygons. Finally, we calculated mean average precision and overall accuracy for each of the models based on Equations 4 and 5. Here, N is the number of total classes:

$$\text{Mean avergage precision} = \frac{1}{N}\sum_{K=1}^{N} \text{average precision for class } K \tag{4}$$

$$\text{Overall accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{total predicted}} \tag{5}$$

Results and Discussion

Models with Different Augmentation Methods

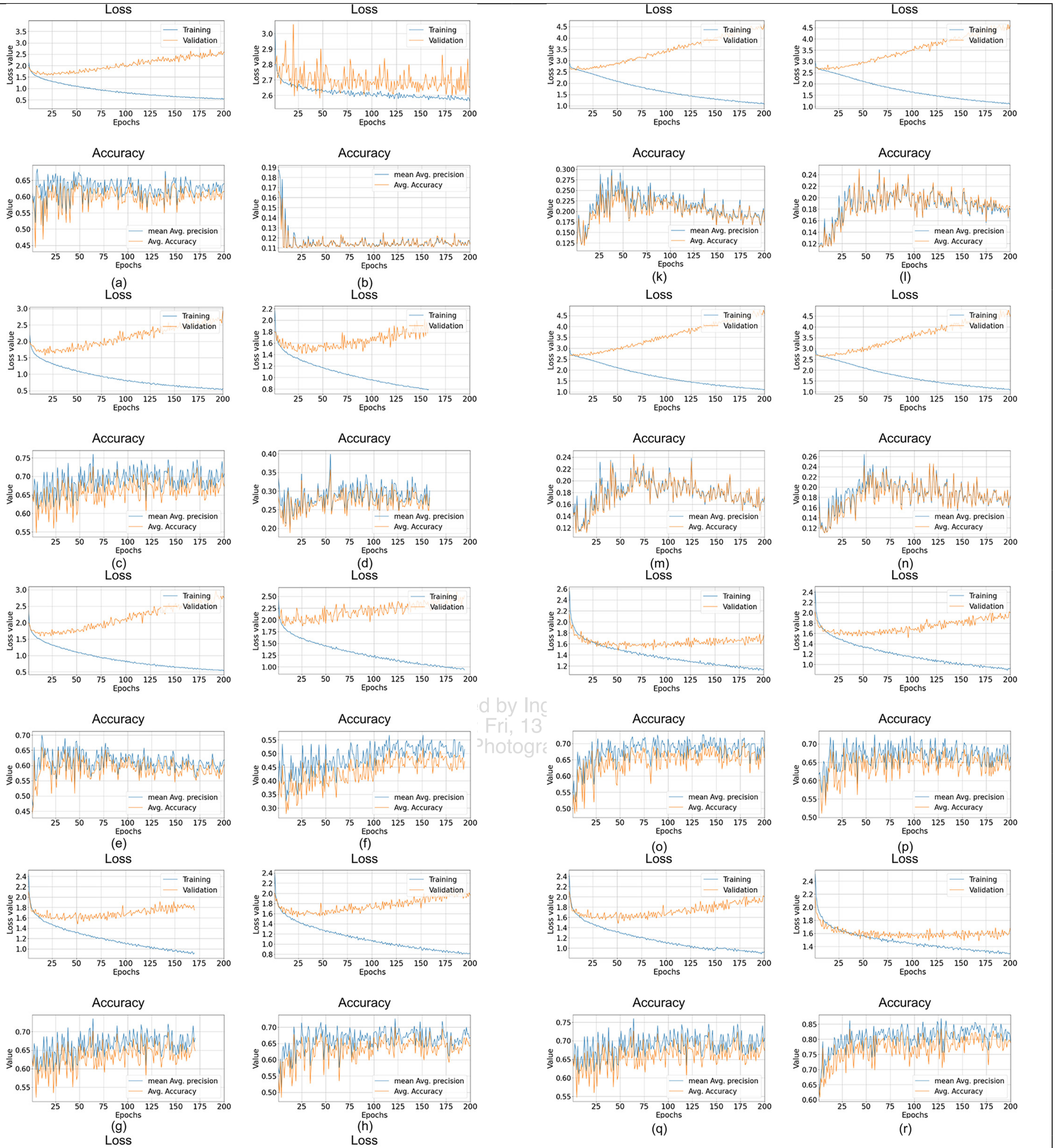
After the model training step was completed, we calculated assessment values for each of the models (Figure 7). Some augmentation methods outperformed the model without any augmentation. However, some augmentation methods did not perform well, as expected. Choosing the best seven methods, we trained another model named the top 7 model and then calculated the assessment values for that model. Figure 7 shows that the top 7 model outperformed the individual models with a 79.6% mAP and 79.3% overall accuracy.

The rotation augmentation methods and the crop method did not perform well compared to other augmentation methods. When the images are cropped, the corners of the images are filled with zero values to match the input image size, and thus the image distribution is very much changed. This could be a reason why rotation methods did not improve the performance. Figure 8 depicts sample outputs from different augmentation methods. Detected polygons are marked in different colors. As we observed in the accuracy plots, certain augmentation methods outperformed in detecting the polygon boundaries.

Among the single augmentation methods, the FlipLR method performed the best; this method does not change the distribution of the input images. However, the salt and pepper noise method also performed well. Salt and pepper noise adds some black and white pixels randomly in the data. The amount of these pixels is not enough to change the distribution widely but is able to mimic digital noise in the image and makes the model robust against noise. As seen on the probability density function and the cumulative distribution function plots (Figure 9), the contributions of the salt and pepper noise in the higher and the lower ends of the possible pixel values are evident.

Models with Separate Tundra Types

We used our trained models on different tundra types and predicted for different tundra types. Table 2 shows the mean average precision for models trained on and predicted for different tundra types. These models performed better when trained and tested on the same tundra types. However, for the model trained on non–tussock sedge (G3) actually performed better on the sedge/grass (W1) tundra type. The reason



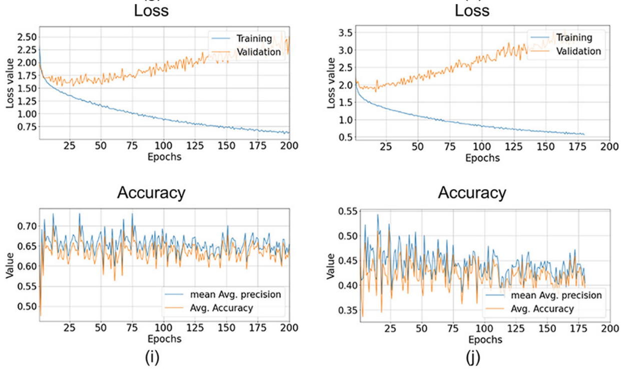


Figure 5. Loss and accuracy plots for different augmentation methods. (a) No augmentation. (b) Crop. (c) FlipLRUD (flip left to right and up and down). (d) FlipLR (flip left to right). (e) FlipUD (flip up and down). (f) Gaussian noise. (g) Hue saturation. (h) Hue. (i) Saturation. (j) Invert. (k) Rotation (30). (l) Rotation (60). (m) Rotation (120). (n) Rotation (150). (o) Spectral. (p) salt and pepper noise. (q) Salt and pepper and FlipLR. (r) Top 7 augmentations.

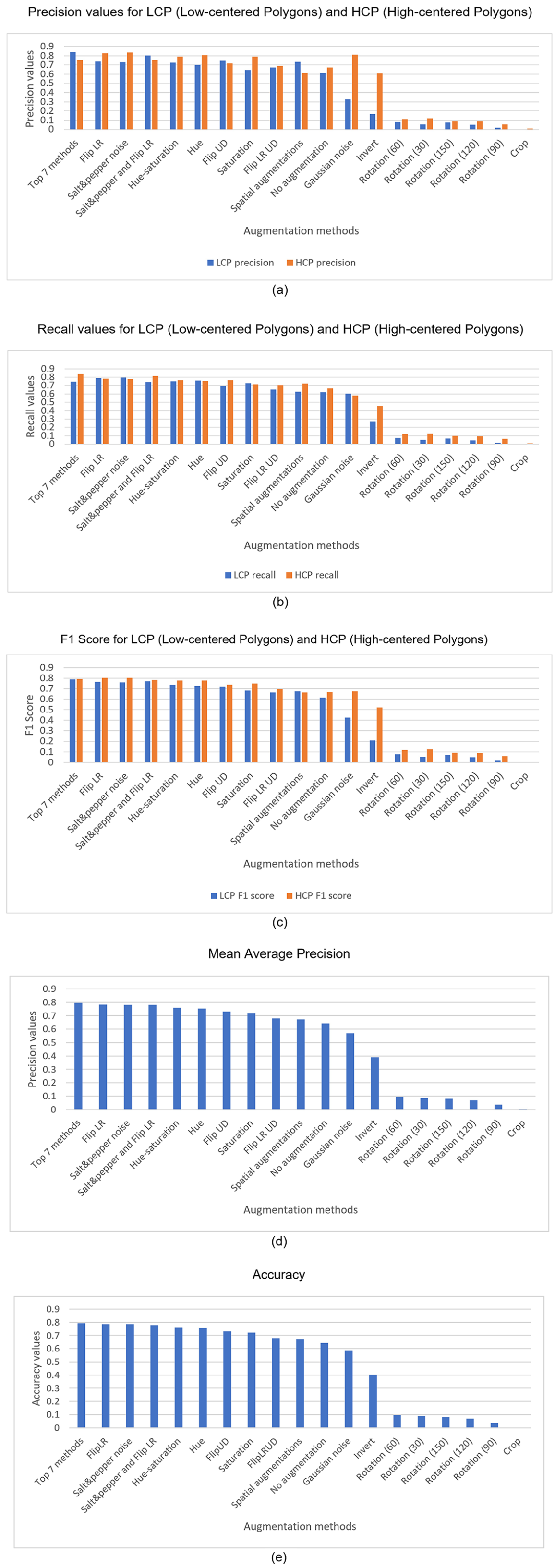


Figure 7. Performance analysis of augmentation methods. (a) Precision values for low-centered polygons (LCP) and high-centered polygons (HCP). (b) Recall values for LCP and HCP. (c) F1 score for LCP and HCP. (d) Mean average precision. (e) Accuracy. FlipLR = flip left to right; FlipUD = flip up and down; FlipLRUD = flip left to right and up and down).

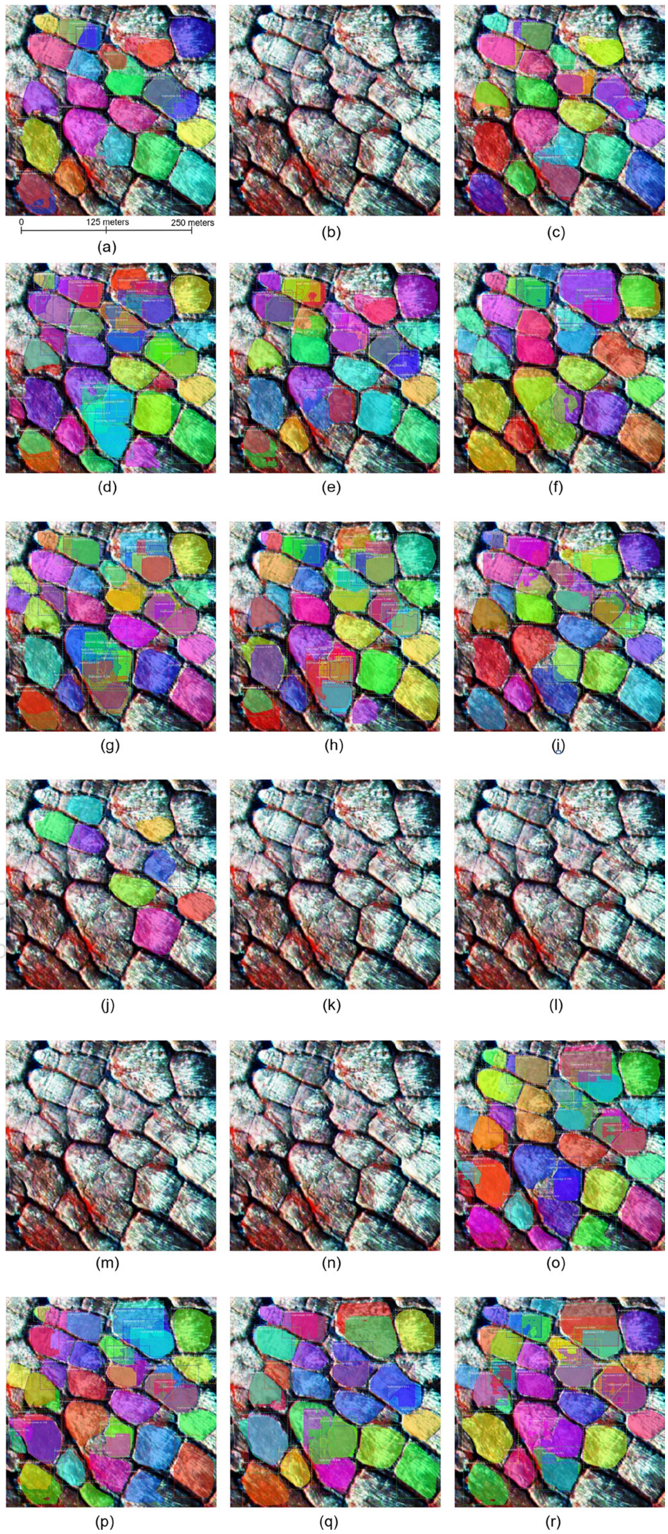


Figure 8. Sample outputs with different augmentation methods. (a) No augmentation. (b) Crop. (c) FlipLRUD (flip left to right and up and down). (d) FlipLR (flip left to right). (e) FlipUD (flip up and down). (f) Gaussian noise. (g) Hue saturation. (h) Hue. (i) Saturation. (j) Invert. (k) Rotation (30). (l) Rotation (60). (m) Rotation (120). (n) Rotation (150). (o) Spectral. (p) Salt and pepper noise. (q) Salt and pepper and FlipLR. (r) Top 7 augmentations.

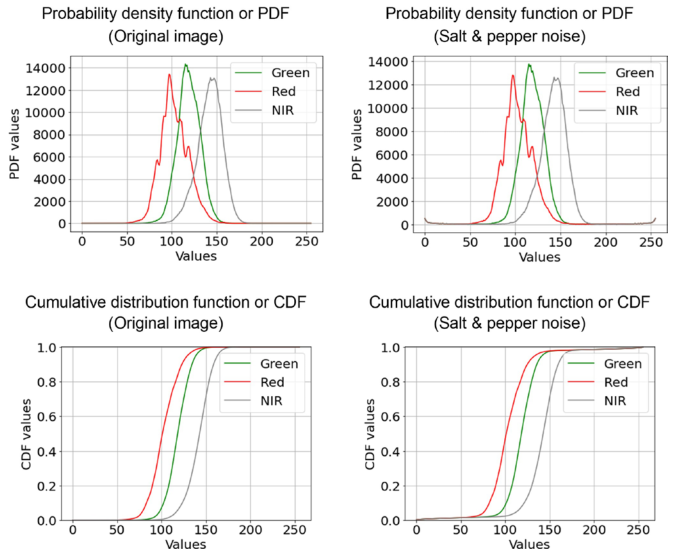


Figure 9. Probability density function (PDF) and cumulative distribution function (CDF) for original image and salt and pepper noise augmented image.

could be the similarity between these two types and the inadequate numbers of polygons of these tundra types.

Table 2. Mean average precision values for models trained on different tundra types: non–tussock sedge (G3), tussock sedge (G4), other tundra types, and sedge/grass (W1).

Mean Average Precision	G3	G4	Other Tundra Types	W1
G3	0.58	0.32	0.47	0.57
G4	0.08	0.79	0.18	0.05
Other tundra types	0.35	0.69	0.62	0.22
W1	0.75	0.35	0.54	0.8

We also predicted the overall accuracy values for the models trained and tested with different tundra types (Table 3). We observed that the models trained and tested on the same tundra types performed better. However, exceptions were found. For example, the model trained with the G3 tundra type performed the best with the W1 tundra type, and the model trained with the G4 tundra type performed the best with other tundra types. This calls for further analysis to better understand the underlying reasons linking the tundra types and model performances.

Table 3. Overall accuracy values for models trained on different tundra types: non–tussock sedge (G3), tussock sedge (G4), other tundra types, and sedge/grass (W1).

Overall Accuracy	G3	G4	Other Tundra Types	W1
G3	0.66	0.13	0.13	0.62
G4	0.08	0.66	0.66	0.05
Other tundra types	0.35	0.79	0.79	0.22
W1	0.75	0.36	0.36	0.8

Conclusion

Mapping IWPs from large satellite imagery requires a huge amount of computational resources as well as large volume of annotated images. We implemented the Mask R-CNN model for segmentation and classification of IWPs from commercially available satellite imagery. We have improved the model performance and found promising results by applying augmentation methods on top of the regular Mask R-CNN model. We explored an array of augmentation methods in the training process. Our results suggested that not all augmentation methods stand as favorable for improving prediction performance. We also trained separate Mask R-CNN models for separate tundra types. The lack of annotated data seems to be visible in the model performance when trained with separate tundra types. Our future research will further investigate the impact of augmentations methods on permafrost feature modeling efforts.

Acknowledgments

This research was supported by the US National Science Foundation grants 1720875, 1722572, 1927872, 1927723, and 1927729. Supercomputing resources were provided by the Extreme Science and Engineering Discovery Environment (award DPP 190001) and Texas Advanced Computing Center (award DPP20001). The authors would like to thank the Polar Geospatial Center at the University of Minnesota for imagery support.

References

Arun Pandian, J., G. Geetharamani, and B. Annette. 2019. Data augmentation on plant leaf disease image dataset using image manipulation and deep learning techniques. Pages 199–204 in *Proceedings of the 2019 IEEE 9th International Conference on Advanced Computing, IACC 2019*. https://doi.org/10.1109/IACC48062.2019.8971580.

Bhuiyan, M. A. E., C. Witharana, and A. K. Liljedahl. 2020. Use of very high spatial resolution commercial satellite imagery and deep learning to automatically map ice-wedge polygons across tundra vegetation types. *Journal of Imaging* 6(12):137. https://doi.org/10.3390/JIMAGING6120137

Carvalho, O. O. de C. J.-R. Sensing, 2021. Instance segmentation for large, multi-channel remote sensing imagery using Mask-RCNN and a mosaicking approach. https://doi.org/10.3390/rs13010039.

Davidson, S. J., M. J. Santos, V. L. Sloan, J. D. Watts, G. K. Phoenix, W. C. Oechel, and D. Zona. 2016. Mapping Arctic tundra vegetation communities using field spectroscopy and multispectral satellite data in north Alaska, USA. *Remote Sensing* 8(12):978. https://doi.org/10.3390/RS8120978.

Elgendi, M., M. U. Nasir, Q. Tang, D. Smith, J.-P. Grenier, C. Batte, B. Spieler, W. D. Leslie, C. Menon, R. R. Fletcher, N. Howard, R. Ward, W. Parker, and S. Nicolaou. 2021. The effectiveness of image augmentation in deep learning networks for detecting COVID-19: A geometric transformation perspective. *Frontiers in Medicine* 8:629134. https://doi.org/10.3389/FMED.2021.629134.

Galdran, A., A. Alvarez-Gila, M. I. Meyer, C. L. Saratxaga, T. Araújo, E. Garrote, G. Aresta, P. Costa, A. M. Mendonça, and A. Campilho. 2017. Data-driven color augmentation techniques for deep skin image analysis. http://arxiv.org/abs/1703.03702

Grm, K., V. Struc, A. Artiges, M. Caron, and H. K. Ekenel. 2018. Strengths and weaknesses of deep learning models for face recognition against image degradations. *IET Biometrics* 7(1):81–89. https://doi.org/10.1049/iet-bmt.2017.0083.

He, K., G. Gkioxari, P. Dollár, and R. Girshick. 2017. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42(2):386–397. https://arxiv.org/abs/1703.06870v3.

He, K., X. Zhang, S. Ren, and J. Sun. 2015. Deep residual learning for image recognition. Pages 770–778 in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, held in 2015. https://arxiv.org/abs/1512.03385v1.

Juutinen, S., T. Virtanen, 2017. Spatial variation and seasonal dynamics of leaf-area index in the arctic tundra-implications for linking ground observations and satellite images. https://doi.org/10.1088/1748-9326/aa7f85.

Karimov, A., A. Razumov, R. Manbatchurina, K. Simonova, I. Donets, A. Vlasova, Y. Khramtsova, and K. Ushenin. 2019. *Comparison of UNet, ENet, and BoxENet for Segmentation of Mast Cells in Scans of Histological Slices*.

Lara, M. J., A. D. McGuire, E. S. Euskirchen, H. Genet, S. Yi, R. Rutter, C. Iversen, V. Sloan, and S. D. Wullschleger. 2020. Local-scale Arctic tundra heterogeneity affects regional-scale carbon dynamics. *Nature Communications* 11(1):1–10. https://doi.org/10.1038/s41467-020-18768-z.

Lateef, F. and Y. Ruichek. 2019. Survey on semantic segmentation using deep learning techniques. *Neurocomputing* 338:321–348. https://doi.org/10.1016/J.NEUCOM.2019.02.003.

Leffingwell, E. de K. 1919. The Canning River region, northern Alaska. https://doi.org/10.3133/PP109.

Liljedahl, A. K., J. Boike, R. P. Daanen, A. N. Fedorov, G. V. Frost, G. Grosse, L. D. Hinzman, Y. Iijma, J. C. Jorgenson, N. Matveyeva, M. Necsoiu, M. K. Raynolds, V. E. Romanovsky, J. Schulla, K. D. Tape, D. A. Walker, C. J. Wilson, H. Yabuki, and D. Zona. 2016. Pan-Arctic ice-wedge degradation in warming permafrost and its influence on tundra hydrology. *Nature Geoscience* 9(4):312–318. https://doi.org/10.1038/NGEO2674.

Liu, N., P. Budkewitsch, and P. Treitz. 2017. Examining spectral reflectance features related to Arctic percent vegetation cover: Implications for hyperspectral remote sensing of Arctic tundra. *Remote Sensing of Environment* 192:58–72. https://doi.org/10.1016/J.RSE.2017.02.002.

Liu, Y. and L. Wu. 2016. Geological disaster recognition on optical remote sensing images using deep learning. *Procedia Computer Science* 91:566–575. https://doi.org/10.1016/j.procs.2016.07.144.

Long, Y., Y. Gong, Z. Xiao, and Q. Liu. 2017. Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing* 55(5):2486–2498. https://doi.org/10.1109/TGRS.2016.2645610

Ma, L., Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson. 2019. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing* 152:166–177. https://doi.org/10.1016/J.ISPRSJPRS.2019.04.015.

Magnússon, R. Í., J. Limpens, J. van Huissteden, d. Kleijn, T. C. Maximov, R. Rotbarth, U. Sass-Klaassen, and M.M.P.D. Heijmans. 2020. Rapid vegetation succession and coupled permafrost dynamics in Arctic thaw ponds in the Siberian lowland tundra. *Journal of Geophysical Research: Biogeosciences* 125(7):e2019JG005618. https://doi.org/10.1029/2019JG005618.

Mahmoud, A., S. Mohamed, 2020. Object detection using adaptive mask RCNN in optical remote sensing images. *International Journal of Intelligent Engineering and Systems*13(1). https://doi.org/10.22266/ijies2020.0229.07.

Mikola, J., T. Virtanen, M. Linkosalmi, E. Vähä, J. Nyman, O. Postanogova, A. Räsänen, D. Johan Kotze, T. Laurila, S. Juutinen, V. Kondratyev, and M. Aurela. 2018. Spatial variation and linkages of soil and vegetation in the Siberian Arctic tundra—Coupling field observations with remote sensing data. *Biogeosciences* 15(9):2781–2801. https://doi.org/10.5194/bg-15-2781-2018.

Paoletti, M. E., J. M. Haut, J. Plaza, and A. Plaza. 2019. Deep learning classifiers for hyperspectral imaging: A review. *ISPRS Journal of Photogrammetry and Remote Sensing* 158:279–317. https://doi.org/10.1016/J.ISPRSJPRS.2019.09.006.

Raynolds, M., D. Walker, A. Balser, 2019. A raster version of the Circumpolar Arctic Vegetation Map (CAVM). https://www.sciencedirect.com/science/article/pii/S0034425719303165.

Rizwan I Haque, I. and J. Neubert. 2020. Deep learning approaches to biomedical image segmentation. *Informatics in Medicine Unlocked* 18:100297. https://doi.org/10.1016/J.IMU.2020.100297.

Romero, A., C. Gatta, and G. Camps-Valls. 2016. Unsupervised deep feature extraction for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing* 54(3):1349–1362. https://doi.org/10.1109/TGRS.2015.2478379.

Shorten, C. and T. M.Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of Big Data* 6(1). https://doi.org/10.1186/S40537-019-0197-0.

Steedman, A., 2016. Spatio-temporal variation in high-centre polygons and ice-wedge melt ponds, Tuktoyaktuk coastlands, Northwest Territories. *Wiley Online Library* 28(1):66–78. https://doi.org/10.1002/ppp.1880.

Stow, D. A., B. H. Burns, and A. S. Hope. 1993. Spectral, spatial and temporal characteristics of arctic tundra reflectance. *International Journal of Remote Sensing* 14(13):2445–2462. https://doi.org/10.1080/01431169308904285.

Sturtevant, C., 2013. Spatial variation in landscape-level CO2 and CH4 fluxes from arctic coastal tundra: Influence from vegetation, wetness, and the thaw lake cycle. *Wiley Online Library* 19(9):2853–2866. https://doi.org/10.1111/gcb.12247.

Su, H., S. Wei, M. Yan, C. Wang, J. Shi, and X. Zhang. 2019. Object detection and instance segmentation in remote sensing imagery based on precise Mask R-CNN. *International Geoscience and Remote Sensing Symposium* 1454–1457. https://doi.org/10.1109/IGARSS.2019.8898573.

Verdonen, M., 2020. Periglacial vegetation dynamics in Arctic Russia: Decadal analysis of tundra regeneration on landslides with time series satellite imagery. https://doi.org/10.1088/1748-9326/abb500.

Waleed Abdulla. 2017. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow. GitHub Repository. https://github.com/matterport/Mask_RCNN.

Xie, F., M. Shi, Z. Shi, J. Yin, and D. Zhao. 2017. Multilevel cloud detection in remote sensing images based on deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10(8):3631–3640. https://doi.org/10.1109/JSTARS.2017.2686488.

Zabawa, L., A. Kicherer, L. Klingbeil, R. Töpfer, H. Kuhlmann, and R. Roscher. 2020. Counting of grapevine berries in images via semantic segmentation using convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing* 164:73–83. https://doi.org/10.1016/j.isprsjprs.2020.04.002.

Zhang, C., I. Sargent, X. Pan, H. Li, A. Gardiner, J. Hare, and P. M. Atkinson. 2019. Joint deep learning for land cover and land use classification. *Remote Sensing of Environment* 221:173–187. https://doi.org/10.1016/J.RSE.2018.11.014.

Zhang, W., C. Witharana, A. Liljedahl, M.K.-R. Sensing, 2018. Deep convolutional neural networks for automated characterization of arctic ice-wedge polygons in very high spatial resolution aerial imagery. https://doi.org/10.3390/rs10091487.

Zhao, Z. Q., P. Zheng, S. T. Xu, and X. Wu. 2019. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems* 30(11):3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865.

Zheng, J., X. Xu, G. Jia, and W. Wu. 2020. Understanding the spring phenology of Arctic tundra using multiple satellite data products and ground observations. *Science China Earth Sciences* 63(10):1599–1612. https://doi.org/10.1007/S11430-019-9644-8.

Zhong, L., L. Hu, and H. Zhou. 2019. Deep learning based multi-temporal crop classification. *Remote Sensing of Environment* 221:430–443. https://doi.org/10.1016/J.RSE.2018.11.032.

Zuo, L., P. He, C. Zhang, and Z. Zhang. 2020. A robust approach to reading recognition of pointer meters based on improved mask-RCNN. *Neurocomputing* 388:90–101. https://doi.org/10.1016/j.neucom.2020.01.032.