

COUNTING ICE-WEDGE POLYGONS FROM SPACE: USE OF COMMERCIAL SATELLITE IMAGERY TO MONITOR CHANGING ARCTIC POLYGONAL TUNDRA

Amit Hasan¹, Mahendra R. Udawalpola¹, Chandi Witharana¹, Anna K. Liljedahl²

¹ Dept. of Natural Resources and the Environment, University of Connecticut, Storrs, Connecticut, USA - (amit.hasan, mahendra.udawalpola, chandi.witharana)@uconn.edu

² Woodwell Climate Research Center, Falmouth, Massachusetts, USA - aliljedahl@woodwellclimate.org

KEY WORDS: Arctic, Ice-wedge polygons, Deep learning, Satellite imagery, Mapping, Mask R-CNN, Augmentations.

ABSTRACT:

The microtopography associated with ice wedge polygons (IWPs) governs the Arctic ecosystem from local to regional scales due to the impacts on the flow and storage of water and therefore, vegetation and carbon. Increasing subsurface temperatures in Arctic permafrost landscapes cause differential ground settlements followed by a series of adverse microtopographic transitions at sub decadal scale. The entire Arctic has been imaged at 0.5 m or finer resolution by commercial satellite sensors. Dramatic microtopographic transformation of low-centered into high-centered IWPs can be identified using sub-meter resolution commercial satellite imagery. In this exploratory study, we have employed a Deep Learning (DL)-based object detection and semantic segmentation method named the Mask R-CNN to automatically map IWPs from commercial satellite imagery. Different tundra vegetation types have distinct spectral, spatial, textural characteristics, which in turn decide the semantics of overlying IWPs. Landscape complexity translates to the image complexity, affecting DL model performances. Scarcity of labelled training images, inadequate training samples for some types of tundra and class imbalance stand as other key challenges in this study. We implemented image augmentation methods to introduce variety in the training data and trained models separately for tundra types. Augmentation methods show promising results but the models with separate tundra types seem to suffer from the lack of annotated data.

1. INTRODUCTION

A network of polygonal pattern 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 (Zhang et al. 2018). Researchers (Leffingwell et al. 1919) described two major types of IWPs: polygons with elevated blocks or high-centered polygons, polygons with depressed blocks or low-centered polygons. In recent years, a dramatic microtopographic transformation of low-centered IWPs into high-centered IWPs across the Arctic tundra region was documented using sub-meter resolution commercial satellite imagery (Steedman et al. 2017) (Liljedahl et al. 2016). Commercially available high resolution satellite imagery is already available, Imagery archives are swiftly rolling to petabyte scale. Yet, the imagery-derived products are rare. 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 implementations of deep learning convolutional neural nets (DLCNNs) in computer vision (CV) applications have received great interest from the remote sensing community. There has been a lot of recent research where researchers have tried to integrate DLCNNs to solve remote sensing classification related problems such as land use and land cover type detection, feature extraction from remote sensing images, etc. Deep learning convolutional neural nets perform well in terms of object detection, image segmentation, semantic object instance segmentation. A lot of DLCNN architectures have been published, trained and tested with different types of imagery. Each of these architectures have their own advantages and disadvantages with respect to the computation time and resources. Previous studies show promising results found by implementing the implementation of deep learning convolutional neural networks with commercial

satellite imagery (Zhang et al. 2018) (Witharana et al. 2020). However, there are scopes of implementing image augmentation techniques with the regular Mask R-CNN model. The Mask R-CNN is an advanced model that does image detection and segmentation at the same time. Image segmentation process detects each of the polygons in our image tiles and the detection pipeline decides whether a polygon is a high-centered or a low-centered IWP. In this study, we have implemented 17 different types of augmentation methods. Some methods are single augmentation methods and some consist of multiple augmentation methods. Based on the type of augmentation method, some changes the image distribution and some keep the distribution of the input image unchanged.

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 (ConvNet/CNN). This can be changed to different types of CNN models, we used the ResNet-50 structure (He et al. 2016) as the backbone. To initialize the model, we have practised the transfer learning approach. In this approach, the model is already trained based on some dataset. Our backbone was pretrained based on the ImageNet dataset. We retrained the Mask R-CNN model with different augmentation methods using our dataset so that the model can be used for the detection and segmentation of the ice wedge polygons.

We noticed that IWPs have different spectral, spatial, textural characteristics based on the tundra vegetation type. For example, the IWPs sampled from the tussock sedge tundra show different characteristics from the IWPs sampled from the non-tussock sedge tundra region. These differences affect the model performance. So, we trained Mask R-CNN models 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 DL CNN method (Mask R-CNN) to characterize the tundra ice-wedge polygon 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.

2. METHODS

2.1 Study Area

We extracted a total of 370 image tiles of varying dimensions (292×292 pixels, 345×345 pixels, 507×507 pixels, 199×199 pixels, etc) out of 3 satellite imagery scenes from the north slope of Alaska and from the Banks Island, Canada (Figure 1). These areas are covered mostly by tussock and non-tussock sedge tundra, sedge/grass moss wetland and other types of tundra. We annotated a total of 12561 polygons (5,620 low-centered and 6,941 high-centered polygons) from these image tiles manually.

We prepared 3 sets of images out of all available images as the training, test and validation datasets. The training dataset was used for training the model and the validation dataset was used to check model performance while training the model. The test dataset was used to calculate the performance of the final model.

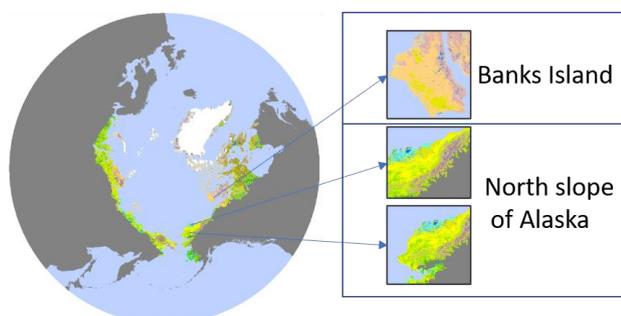


Figure 1. Circumpolar arctic vegetation map (CAVM) (Raynolds et al. 2019) showing the study area.

2.2 Model Architecture

We implemented a deep-learning based model named as the Mask R-CNN (He et al. 2017) (Figure 2). This model is specialized in object detection as well as instance segmentation at the same time.

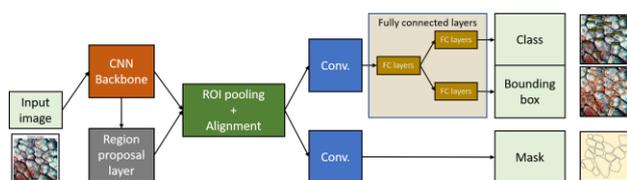


Figure 2. Simplified block diagram of the Mask R-CNN model.

Our codes are developed based on an open-source package built on Keras and Tensorflow developed by the team of Mask R-CNN on Github (https://github.com/matterport/Mask_RCNN). The model consists of a CNN backbone, region proposal network, neural networks for predicting classes, bounding box and generating the mask. We have used ResNet-50 (He et al. 2016) pretrained with the ImageNet dataset 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.

2.3 Augmentation Methods

Image augmentation is a process that modifies training images in a variety of ways and act like additional training images to the model. Image augmentation, thus, can boost the performance of deep learning models by introducing additional training data. In the Mask R-CNN model, we have the ability to introduce augmentation methods. Table 1 exhibits the augmentation methods that we used in our study.

Augmentation methods	Description	Names used in accuracy plots
Crop	generates smaller-sized subimages from given full-sized input images	Crop
FlipLR	flips the image horizontally	FlipLR
FlipUD	flips the image vertically	FlipUD
FlipLRUD	combination of FlipLR and FlipUD	FLRUD
Gaussian noise	Adds noise sampled from gaussian distributions	GN
Hue	multiplies the hue of images by random values	Hue
Saturation	multiplies the saturation of images by random values	Sat
Hue-saturation	multiplies the hue and saturation of images by random values	HS
Invert	subtracts all pixel values from 255	Invert
Rotation (x)	Apply affine rotation of 'X' degrees on the y-axis to input data	RX (ex: R30, R60)
Salt & pepper noise	adds salt and pepper noise (noisy white-ish and black-ish pixels) to rectangular areas within the image	SPN
Salt & pepper and FlipLR	a combination of salt & pepper noise and FlipLR method	SPFLR
Spectral	a sequential combination of salt & pepper noise, hue, saturation and hue-saturation augmentation methods	Spectral
Top 7 augmentations	a sequential combination of the top 7 augmentation methods based on their mean average precision score on the test dataset	Top7

Table 1. Augmentation methods used in this study.

Some augmentation methods (for example: flipping) do not change the distribution of the input images where some augmentation methods (for example: gaussian noise) change the 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 some combined augmented methods. For example, we have combined the salt & pepper noise and hue augmentation, saturation augmentation, hue-saturation augmentation methods into a single pipeline and named it as spectral augmentation to get the benefits out of all the individual

augmentation methods. We also used a sequential combination of the salt & Pepper noise augmentation and the FlipLR augmentation method. The last augmentation method named as Top7 includes 7 augmentation methods that appeared in the top when ranked by their performance. The performance measurement process is discussed in the accuracy assessment section. Figure 3 lists some of the sample images showing the effects of augmentation methods compared to the original image.

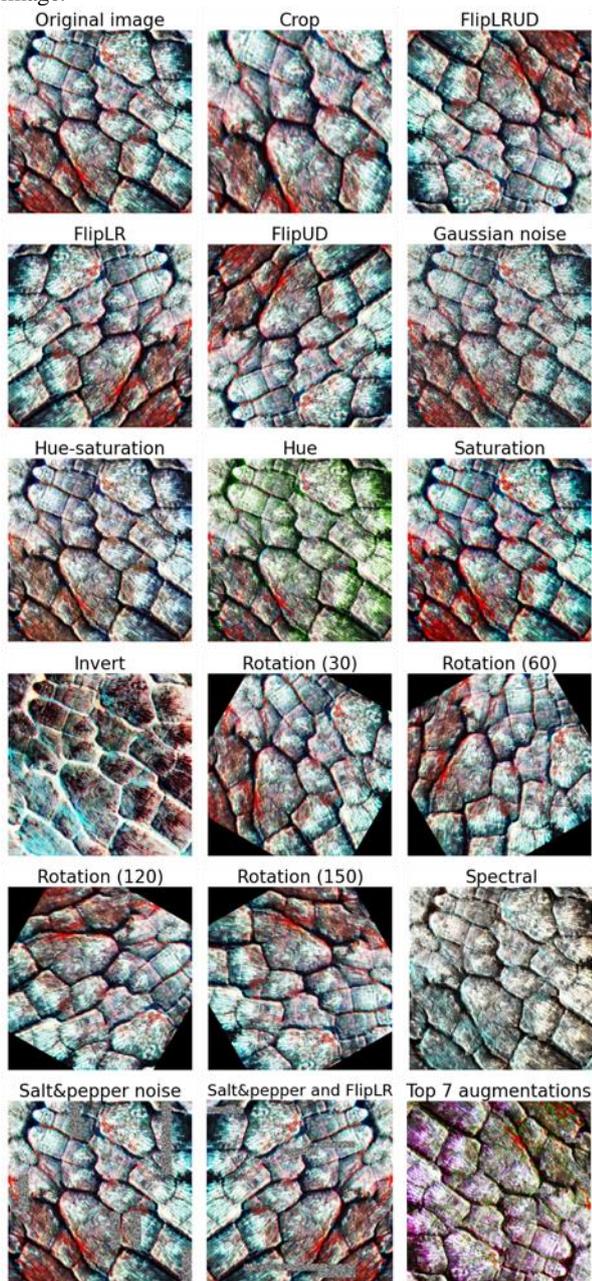


Figure 3. Augmented images compared to the original image.

2.4 Separate Models for Separate Tundra Types

We have seen that different tundra vegetation types have distinct spectral, spatial, textural characteristics, which in turn decide the semantics of overlying IWPs. 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. For this we selected the best augmentation methods based on their

performance, and then train separate models with separate training data sets, each of which will contain only one particular types of tundra. As we have seen the distribution of tundra types in our data compared to the entire arctic, we have seen that our data has a different distribution than the original arctic (Figure 4). However, 3 of the major tundra types cover more than 70% of our sampled data set. So, we have prepared 4 tundra types named as (G3) Non tussock sedge, (G4) Tussock sedge, (W1) Sedge/ grass, and (Others) other tundra types.

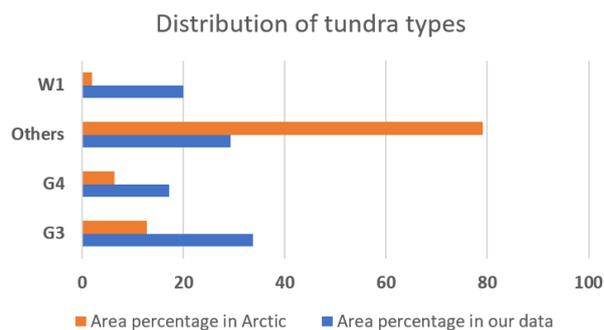


Figure 4. Distribution of tundra types.

2.5 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 ImageNet dataset. The training process was completed in a local machine with Intel(R) Core(TM) i9 CPU with NVIDIA GeForce RTX 2070 SUPER with 8GB of GPU memory. The training time was not measured as multiple training process were run on the local machine at the same time and based on the GPU load the training time varied.

After deciding the augmentation methods and the tundra types, we trained the Mask R-CNN model with mini-batches (we changed the steps size and batch size based on the memory available in the GPU), learning rate of 0.001, learning momentum of 0.9, and weight decay of 0.0001. We had a total of 257 training image tiles (4,019 low-centered polygons and 4,007 high-centered polygons), 53 validation image tiles (773 low-centered polygons and 989 high-centered polygons) and 59 test image tiles (828 low-centered polygons and 945 high-centered polygons).

To optimize the model, we looked into different losses, such as (a) L1 loss (this defines box regression on object detection systems, which is less sensitive to outliers, than other regression loss); (b) Mask R-CNN bounding box loss (this loss indicates the difference between predicted bounding box correction and true bounding box); (c) Mask R-CNN classifier loss (this loss estimates difference of class labels between prediction and ground truth); (d) mask binary cross-entropy loss (this loss measures the performance of a classification model by observing predicted class and actual class); (e) RPN bounding box loss (this loss identifies the regression loss of bounding boxes only when there is object and); (f) 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) or 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, G3 tundra type seem yet to converge when trained for 200 epochs.

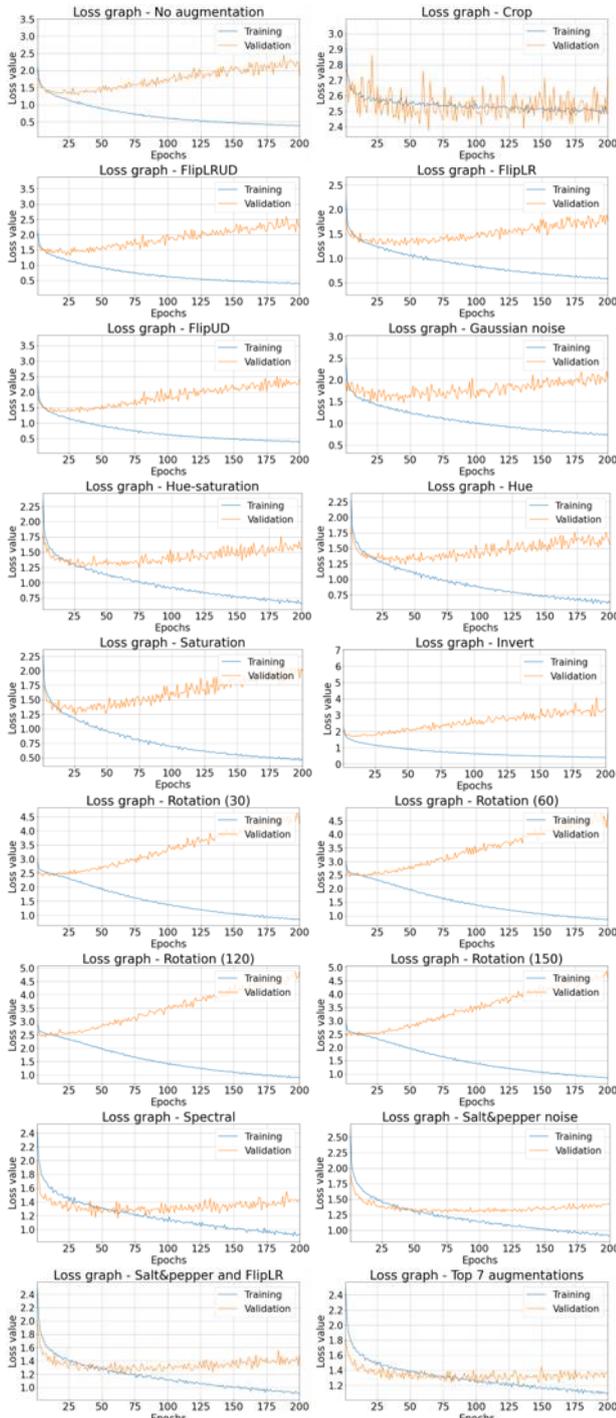


Figure 5. Loss graphs for different segmentation methods.

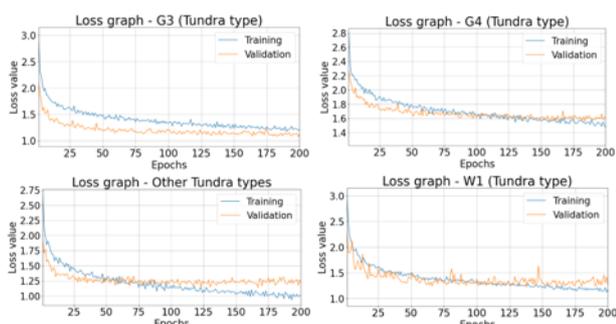


Figure 6. Loss graphs for different tundra types.

2.6 Accuracy Assessment

We conducted a multi-step accuracy assessment for the outputs. The outputs are in the form of class names and binary masks. We calculated the intersection over union (IoU) for each of the polygons in the outputs that matched with the polygon classes in the test dataset. 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.

$$Precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (1)$$

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (2)$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

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.

$$mean\ Avg.Precision = \frac{1}{N} \sum_{K=1}^N Avg\ precision\ for\ class\ K \quad (4)$$

Here, N is the number of total classes.

$$Overall\ Accuracy = \frac{true\ positive + true\ negative}{total\ predicted} \quad (5)$$

3. RESULTS AND DISCUSSION

3.1 Models with Different Augmentation Methods

After the model training step was completed, we have 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. Choosing the best 7 methods, we trained another model named as the Top7 model and then calculated the assessment values for that model.

Figure 7 shows that Top7 model outperformed the individual models with a 79.6% mAP and 79.3% overall accuracy.

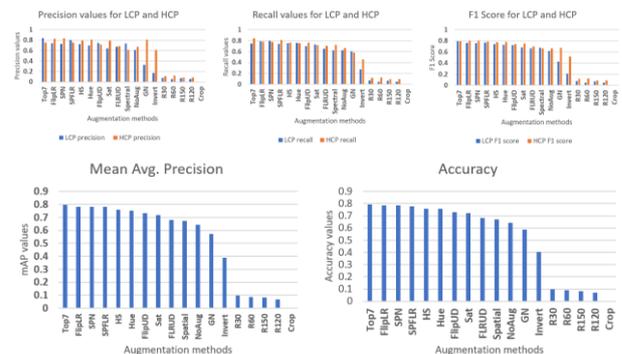


Figure 7. Performance analysis of augmentation methods.

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. Here are some sample outputs from different augmentation methods. Detected polygons are marked in different colors. As we observed in the accuracy plots, some methods did well in detecting the polygon boundaries.

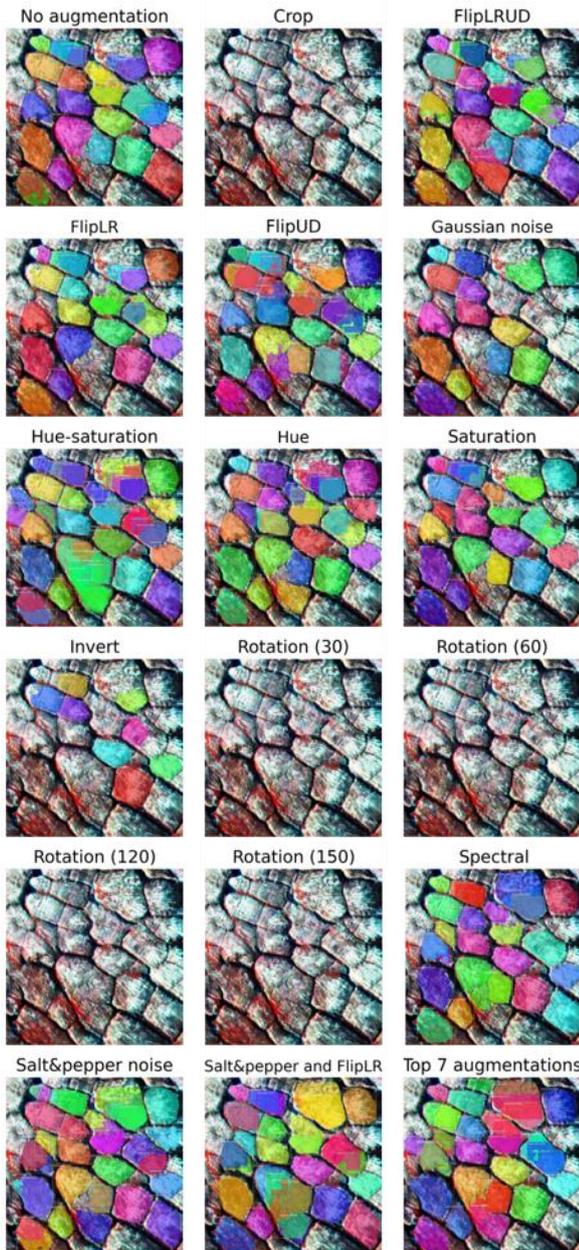


Figure 8. Sample outputs with different augmentation methods.

Among the single augmentation methods, FlipLR method performed the best, this method doesn't change the distribution of the input images. However, the salt & pepper noise method also performed well. Salt & pepper noise adds some black and white pixels randomly in the data. The amount of these pixels are not enough to change the distribution widely, but able to mimic digital noise in the image and makes the model robust against noise. As we can see on the probability density function

and the cumulative distribution function plots, we can see that the contributions of the salt & pepper noise in the higher and the lower end of the possible pixel values.

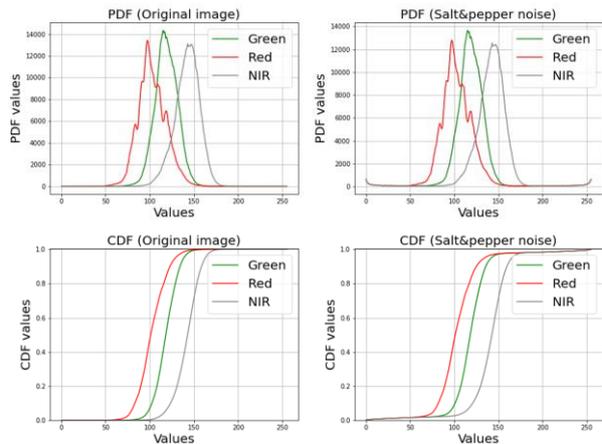


Figure 9. PDF and CDF for original image and salt & pepper noise augmented image.

3.2 Models with Separate Tundra Types

We used our trained models on different tundra types and predicted on different tundra types. Table 2 shows the mean average precision for models trained and predicted on 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 Sedge/grass (W1) tundra type. The reason could be the similarity between these two types and also the inadequate numbers of polygons of these tundra types.

Mean Average Precision	Trained on			
	(G3) Non tussock sedge	(G4) Tussock sedge	other tundra types	(W1) Sedge/grass
(G3) Non tussock sedge	0.58	0.32	0.47	0.57
(G4) Tussock sedge	0.08	0.79	0.18	0.05
other tundra types	0.35	0.69	0.62	0.22
(W1) Sedge/grass	0.75	0.35	0.54	0.8

Table 2. Mean average precision values for models trained on different tundra types.

We also predicted the overall accuracy values for the models trained and tested with different tundra types (Table 3). We saw that the models trained and tested on the same tundra types performed better. However, exceptions were also found. For example: the model trained with G3 tundra type performed the best with W1 tundra type, the model trained with G4 tundra

type performed the best on other tundra types. We will analyse more on the tundra types in our future works.

Overall Accuracy	Trained on				
	(G3) Non tussock sedge	(G4) Tussock sedge	other tundra types	(W1) Sedge/grass	
Predicted on	(G3) Non tussock sedge	0.66	0.13	0.13	0.62
	(G4) Tussock sedge	0.08	0.66	0.66	0.05
	other tundra types	0.35	0.79	0.79	0.22
	(W1) Sedge/grass	0.75	0.36	0.36	0.8

Table 3. Overall accuracy values for models trained on different tundra types.

4. CONCLUSION

Mapping ice wedge polygons from large satellite imagery takes a lot of computational resources as well as a lot of annotated images. We implemented the Mask R-CNN model for segmentation and classification of the ice wedge polygons from commercially available satellite imagery. We have improved the model performance and found promising results by applying augmentation methods on top of regular Mask R-CNN models. We explored many types of augmentation methods in our training process. However, there are many other augmentation methods yet to be explored. We will look further into this topic in our future works. 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. We will continue our analysis based on the tundra types with more image tiles.

ACKNOWLEDGEMENTS

This work is funded by the U.S. National Science Foundation, NSF, Arctic System Science Program (Award #s. 1720875, 1722572, 1720875), Navigating New Arctic Program (Award #s. 1927872, 1927723, 1927729).

REFERENCES

Raynolds M.K., Walker D.A., Balser A., Bay C., Campbell M., Cherosov M.M., Daniëls F.J., Eidesen P.B., Ermokhina K.A., Frost G.V., Jędrzejek B. A. 2019. raster version of the Circumpolar Arctic Vegetation Map (CAVM). *Remote Sensing of Environment* 232111297.2019.

K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.

K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer

Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

Abdulla and Waleed. Mask R-CNN for Object Detection and Instance Segmentation on Keras and TensorFlow. Available online: https://github.com/matterport/Mask_RCNN (accessed on 14 April 2021).

Bhuiyan, M.A.E., Witharana, C. and Liljedahl, A.K., 2019. Big Imagery as a Resource to Understand Patterns, Dynamics, and Vulnerability of Arctic Polygonal Tundra. *AGUFM*, pp.C13E-1374.

Kanevskiy, M.; Shur, Y.; Jorgenson, T.; Brown, D.R.N.; Moskalenko, N.G.; Brown, J.; Walker, D.A.; Raynolds, M.K.; Buchhorn, M. 2017. Degradation and stabilization of ice wedges: Implications for assessing risk of thermokarst in northern Alaska. *Geomorphology*, 297, 20–42.

Liljedahl, A.K.; Boike, J.; Daanen, R.P.; Fedorov, A.N.; Frost, G.V.; Grosse, G.; Hinzman, L.D.; Iijma, Y.; Jorgenson, J.C.; Matveyeva, N.; Necsoiu, M. 2016. Pan-Arctic ice-wedge degradation in warming permafrost and its influence on tundra hydrology. *Nat. Geosci.*, 9, 312.

Muster, S.; Heim, B.; Abnizova, A.; Boike, J. 2013. Water body distributions across scales: a remote sensing based comparison of three arctic tundra wetlands. *Remote Sens.*, 5, 1498–1523.

Skurikhin, A. N.; Wilson, C. J.; Liljedahl, A.; Rowland, J. C. 2014. Recursive active contours for hierarchical segmentation of wetlands in high-resolution satellite imagery of arctic landscapes. *Southwest Symposium on Image Analysis and Interpretation*, 137–140.

Steedman, A. E.; Lantz, T. C.; Kokelj, S. V. 2017. Spatio-temporal variation in high-centre polygons and ice-wedge melt ponds, Tuktoyaktuk coastlands, Northwest Territories. *Permafrost Periglacial Processes*, 28, 66–78.

Zhang, W.; Witharana, C.; Liljedahl, A.K.; Kanevskiy, M. 2018. Deep Convolutional Neural Networks for Automated Characterization of Arctic Ice-Wedge Polygons in Very High Spatial Resolution Aerial Imagery. *Remote Sens.*, 10, 1487.

Leffingwell, E.d.K. The Canning River region, Northern Alaska. U.S. Geological Survey Professional Paper 109. Available online: <https://pubs.usgs.gov/pp/0109/report.pdf> (accessed on 14 April 2021).