Predicting Ice Sheet Surface Velocities in Western Greenland from Ice Sheet Geometry Using a Convolution Neural Network

Aidan Stansberry

Machine Learning

A convolutional neural network UNet was used to predict surface velocities of a portion of the Western Greenland Ice Sheet. In general, the network performed poorly on the test set, with accuracies ranging from 25 to 45 percent. That being said, the network was able to predict where the margin of the ice sheet was, and was also able to predict a general trend of decreasing ice velocity toward the interior of the ice sheet.

Introduction

Relatively recently large amounts of data for ice sheet geometry have become readily available for general use. Machine learning algorithms have also become much more accessible recently as well. Some research (Mohajerani et al 2019, Zhang et al 2019) has used convolutional neural networks to detect calving fronts of sea terminating glaciers, but overall, there is a lot of room for new applications of machine learning to ice sheet problems.

Ice flow models using higher order physics are used to determine velocities of glaciers and ice sheets (eg Morlighem et al 2010, Le Meur et al 2004). It came to mind that the plethora of data available may be useful in predicting the the surface velocity of the ice sheet, without having to revert to higher order numerical models. A convolutional neural network may be able to be trained over the ice sheet and then once trained, could be used to efficiently predict surface velocities based on input data images. I used surface elevation, bed elevation, ice thickness, and water routing data to predict surface velocity maps of the ice sheet. These data sets provide all of the information needed for some physical measures that influence ice sheet velocity such as driving stress. The data sets are fed into a Pytorch Implementation of UNet, (adapted from github.com/milesial/Pytorch-UNet), in order to produce predictions of surface velocity.

Data

Bed elevation, surface elevation, and ice thickness data used are from IceBridge, BedMachine Greeenland Version 3 (Morlighem 2017). This data set provide 150 m x 150 m spatial resolution data set for the entire Greenland ice sheet. The true resolution is at between 150 m and 5 km Data compiled to make this data set were collected between 01 January 1993 and 31 December 2016. The final input data set, which is likely not too useful in the current application form but may be useful in the future is a routing dataset provided by Toby, which was derived from the BedMachine data.

For training and validation velocities, MEaSUREs Greenland Ice Sheet Velocity Map from InSAR Data, Version 2 (Joughin et al 2015) are used. These data are either 500 m or 200 m resolution. I specifically chose a 200 m resolution yearly velocity image and then reprojected it onto a 150 m grid (causing some extra uncertainty in the dataset) for comparison to the 150 m resolution data from BedMachine.

Once the datasets were compiled (Figure 1), the data was cropped into a 180 km north by 160 km east image with the point 100 km north 5 km east at the farthest reach of Issunguata Sermia in western Greenland. The datasets were then cropped into twenty smaller images sliced along the east-west line. Ten were used as training data, and the other ten were used as test data.

Methods

This problem aims to ingest four images of data, and output one image of velocity data. To do this I used UNet, a convolutional neural network that instead of classifying a specified image, outputs a number of images (of the same dimension as the input) equal to the the number of classes a pixel value can be. Each of these outputs represents a probability of the pixels in the image belonging to a specified class. Thus taking the max output of the images gives a guess as to the actual value of each pixel in an image. This image compared to the actual image is used to calculate a loss item to train the network using cross entropy loss.

As mentioned before the output of the network is represented by an integer number of discrete classes. This means, that in order to use this network to predict a surface velocity map, the velocity data (represented by continuous values) had to be binned into discrete classes. The results presented in this paper were binned into 50 bins, each representing around 9 m/yr of velocity. In other words, the resolution of the velocity map produced by the network is 9 m/yr. This choice was somewhat arbitrary, and the data could easily be binned with a higher or lower resolution, although the using more bins took the network longer to reach a minimum.

Once the velocity data was binned, the network was trained for 1000 epochs using the 10 training images (59 by 1085 pixels) at which point there was not considerable improvement to the accuracy of the network.

Results/Discussion

The training accuracy on the set was at least 94 percent for each image in the training set. Figure 2 shows an example of a training image. The place where the algorithm performed the worst, is, not surprisingly, at the boundary between the contour regions. In general these regions provided

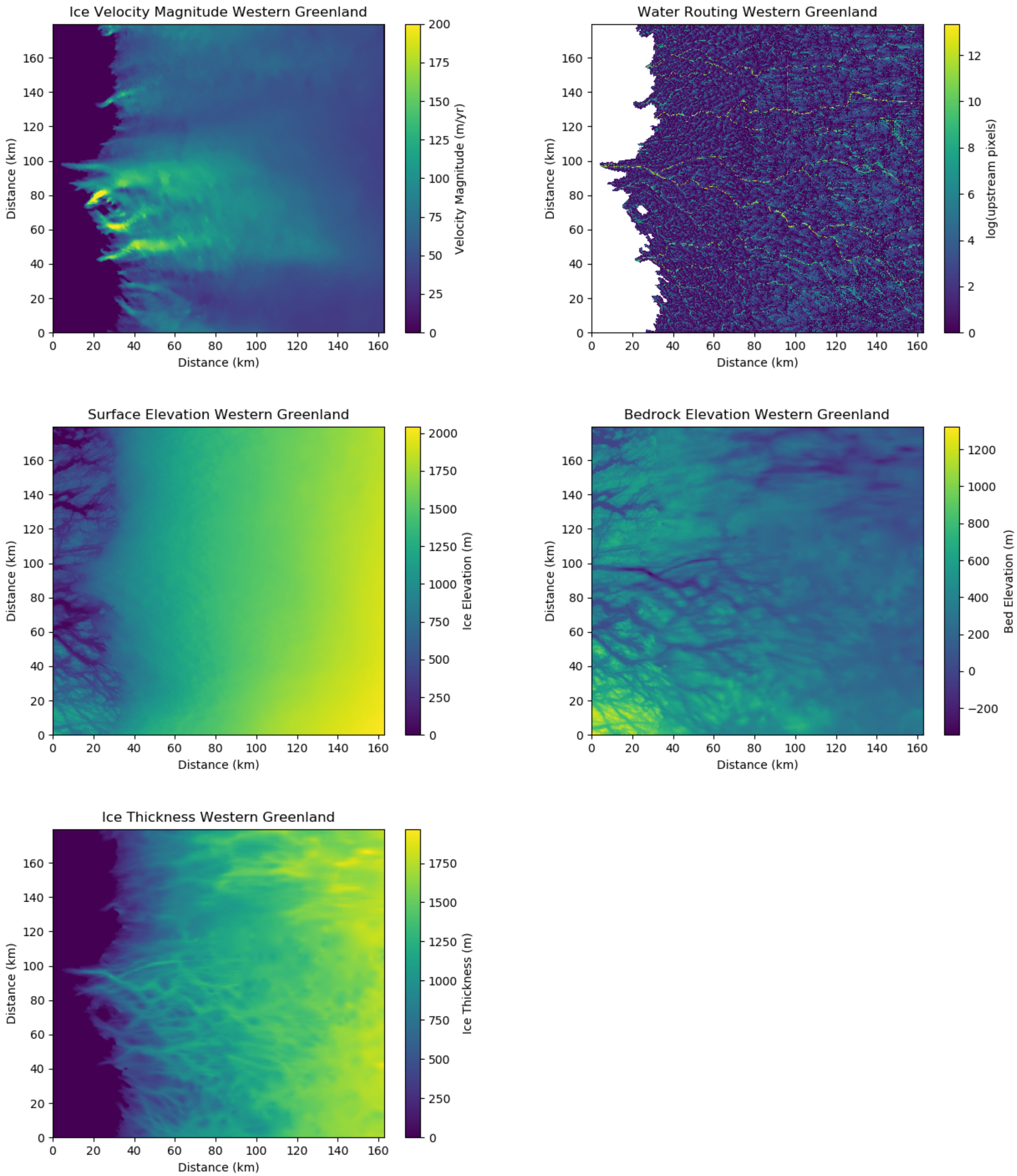

Figure 1: Data sets used

the most model uncertainty. The only other case I saw was a region in a different training image with a particularly high velocity. The network had a difficult time reproducing that signal. Overfitting of the data seems entirely possible given that no regularization or dropout was added to the network. That being said, it is quite fun to see that the image can be reproduced from four completely different sets of data.

The test data accuracy ranged from around 25 to 45 percent with the network. Figure 2 shows an example of an image produced by the test set. The network in general did two things well. First, it is very good at locating the margin of the ice sheet. This isn't too surprising considering that the velocity is zero only when the ice thickness is zero. Ideally the network should be able to recognize this and apparently does. The second, and perhaps more subtle observation, is that the network tends to know that the ice sheet velocity decreases as you move farther towards the interior of the ice sheet. In other words it tended to predict faster velocities towards the the margin of the ice sheet, and slower velocities near to the margin. Every once in a while some of the velocity structures looked somewhat similar to the actual, but these are less prominent in the data and possibly mostly my brain trying to find any pattern it can.

One final note of moderate interest. When the network incorrectly labeled a region, it was more likely to predict a value near to the actual value. Figure 3 shows the number of guesses given plotted versus the difference between the actual and predicted. This could mean that the network may be doing a little better than it appears from the set accuracies. An alternative explanation is that since the majority of velocities on the ice sheet fall between 50 and one hundred, guesses with this region tend to be a safe bet for the network. The result is that guesses are too far off from the actual. The true hope is that the reason this is occurring is that the network recognizes that features that produce a certain velocity are most similar to the conditions that produce a velocity that is closest to its value.

The fact that the neural network was so much better at predicting the training set than the test set is likely a symptom of overfitting and the limited nature of the input datasets. Regularization schemes could be used to try to improve the network or training the network on a larger region of the ice sheet, but I think that the the secondary issue of the input data (the experimental setup) sets is likely more problematic. One very obvious aspect of the dataset is the lack of knowledge of basal traction. This would encompass knowledge of things such as whether the the ice sheet is underlain by soft sediments or hard bedrock. This knowledge could help the network decide how much the ice sticks to the underlying bedrock. A secondary issue is the distribution of water at the bed. The routing data gives some indication of this, but only works assuming that the drainage system is well connected. I could continue, but ultimately this is to say that ice dynamics are complicated and variable from year to year (since this is only a snapshot of the velocity), so it may have been a bit too much to ask this network to solve this problem.

Conclusion

Predicting the surface velocity of the ice sheet using basic ice sheet geometry does not appear to be feasible without either augmenting the used datasets, or adding other datasets. Future studies should consider this when approaching this problem. That being said, the network was able to predict where the margin of the ice sheet was as well as the general trend of decreasing velocity as you move towards the interior of the ice sheet.

One followup studie that could be interesting would be flipping this problem around to try and guess ice thickness given the other parameters. Also, if I had more time to continue working with this network, it would be interesting to see how it performs with more or less bins. It seems possible, that the network might be able predict more generalized velocity fields (slow medium fast or something) with greater accuracy. I would also be interesting in training the network with more random slices of the ice sheet. For example I would want to see if the network performs differently given small squares of the ice sheet instead of long transects.

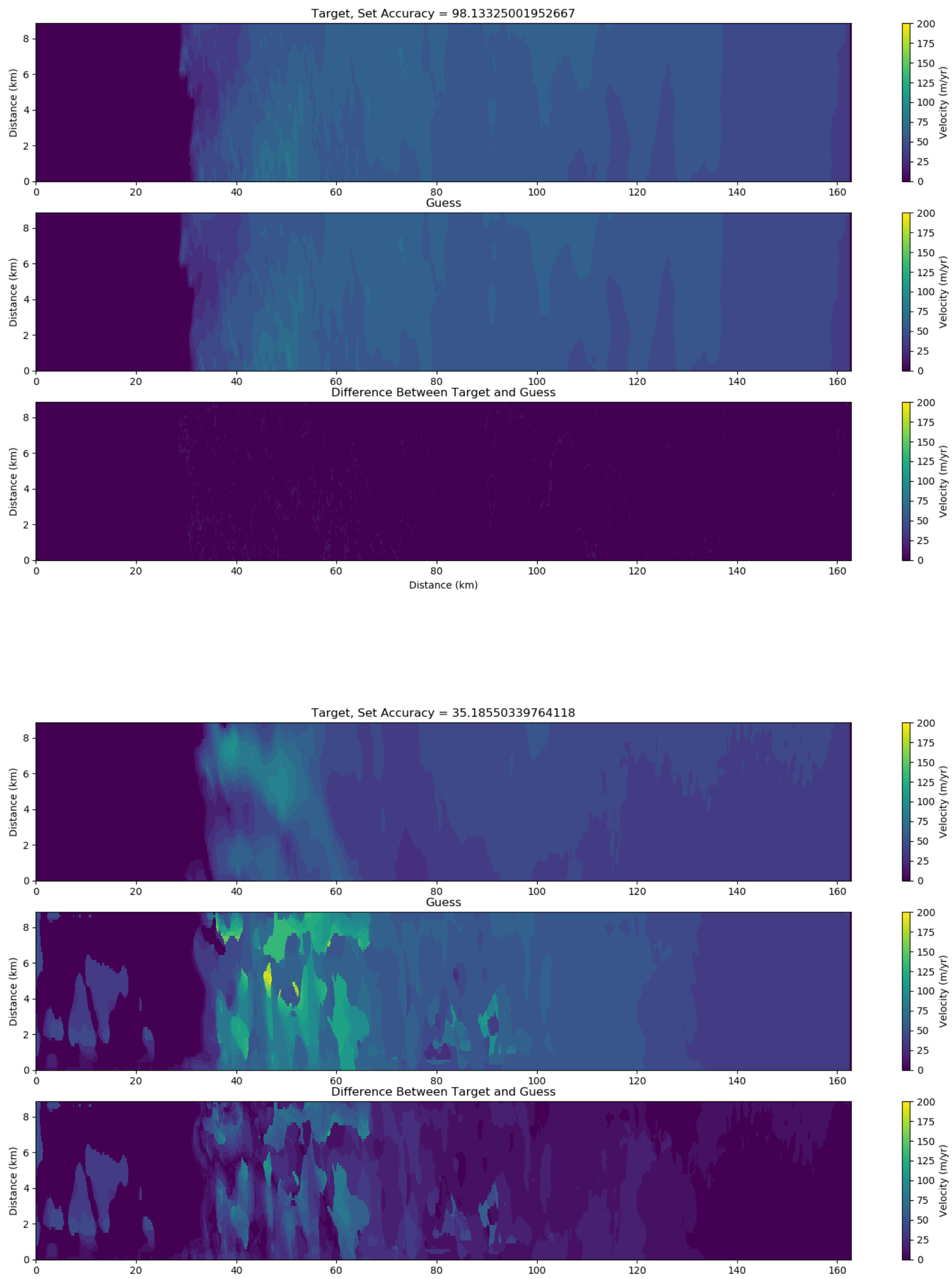

Figure 2: An example from the training (top) and test (bottom) set. The top image shows the INSAR velocity map, the middle image shows the prediction from UNet provided Surface Elevation, Bed Elevation, Routing, and Ice Thickness. The bottom images shows the difference between the predicted and actual. All plots are on the same scale.

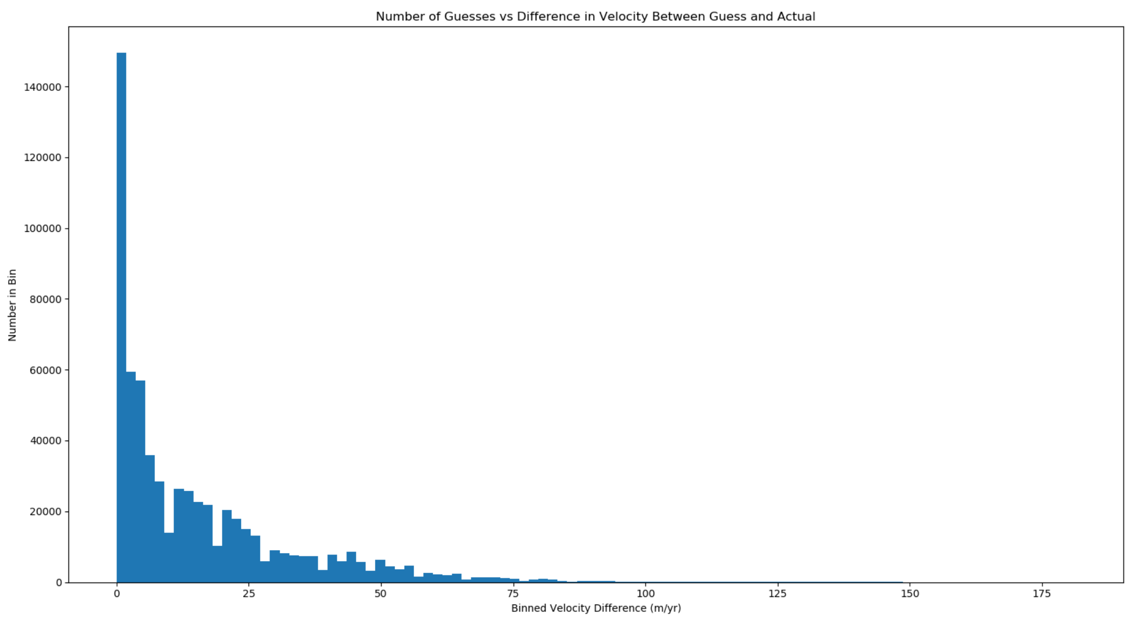

Figure 3: The number of guesses of a specific difference from the actual for each of the test sets (note the higher than 50 bin number, this was from a run with more classes used to train the network. The trend is the same with more or less bins.)

References

1. Morlighem, M. et al. 2017, updated 2018. IceBridge BedMachine Greenland, Version 3. Boulder, Colorado USA. NASA National Snow and Ice Data Center Distributed Active Archive Center. doi: https://doi.org/10.5067/2CIX82HUV88Y.

2. Morlighem, M., C. Williams, E. Rignot, L. An, J. E. Arndt, J. Bamber, G. Catania, N. Chauch, J. A. Dowdeswell, B. Dorschel, I. Fenty, K. Hogan, I. Howat, A. Hubbard, M. Jakobsson, T. M. Jordan, K. K. Kjeldsen, R. Millan, L. Mayer, J. Mouginot, B. Nol, C. O'Cofaigh, S. J. Palmer, S. Rysgaard, H. Seroussi, M. J. Siegert, P. Slabon, F. Straneo, M. R. van den Broeke, W. Weinrebe, M. Wood, and K. Zinglersen. 2017. BedMachine v3: Complete bed topography and ocean bathymetry mapping of Greenland from multi-beam echo sounding combined with mass conservation, Geophysical Research Letters. 44. . https://doi.org/10.1002/2017GL074954

3. Joughin, I., B. Smith, I. Howat, and T. Scambos. 2015, updated 2018. MEaSUREs Greenland Ice Sheet Velocity Map from InSAR Data, Version 2. Boulder, Colorado USA. NASA National Snow and Ice Data Center Distributed Active Archive Center. doi: https://doi.org/10.5067/OC7B04ZM9G6Q.

4. Joughin, I., B. Smith, I. Howat, T. Scambos, and T. Moon. 2010. Greenland Flow Variability from Ice-Sheet-Wide Velocity Mapping, Journal of Glaciology. 56. 415-430.

5. Le Meur, Emmanuel, et al. "Glacier Flow Modelling: A Comparison of the Shallow Ice Approximation and the Full-Stokes Solution." Comptes rendus - Physique 5.7 (2004): 709-22. CrossRef. Web.

6. Mohajerani, Yara, et al. "Detection of Glacier Calving Margins with Convolutional Neural Networks: A Case Study." Remote Sensing 11.1 (2019): 74. CrossRef. Web.

7. Morlighem, M., et al. ”Spatial Patterns of Basal Drag Inferred using Control Methods from a fullStokes and Simpler Models for Pine Island Glacier, West Antarctica.” Geophysical Research Letters 37.14 (2010): L14502,n/a. Research Library Prep. Web.

8. Zhang, Enze, Lin Liu, and Lingcao Huang. ”Automatically Delineating the Calving Front of Jakobshavn Isbr from Multitemporal TerraSAR-X Images: A Deep Learning Approach.” The Cryosphere 13.6 (2019): 1729-41. CrossRef. Web.