

Extracting Rock Outcrop from Antarctic Landsat Imagery using Semantic Segmentation

Shubhi Agarwal
Sam Elkind
Divyanshu Goyal
Ginni Kakkar
Shubhangi Upasani

Group 8

Motivation and Problem Statement

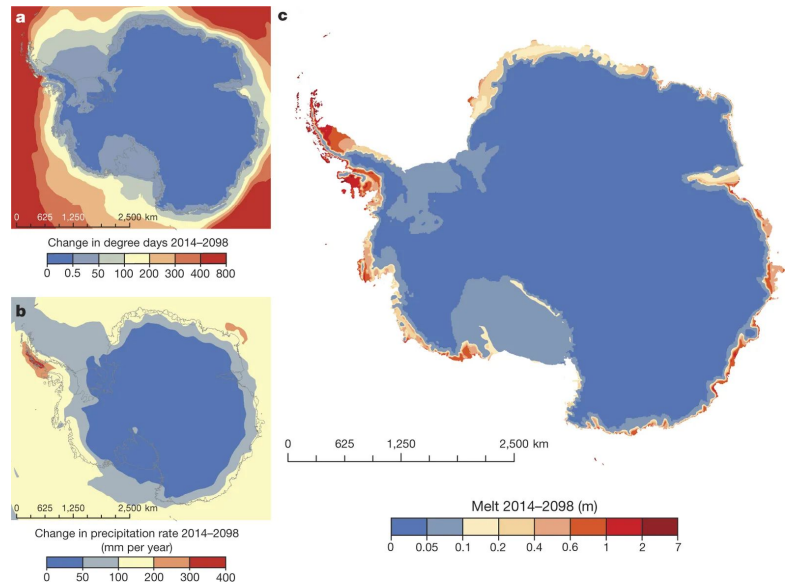
Study and research of Antarctica is important and can help us to analyze:

- Effects of global warming on ice sheets
- Ice sheet dynamics affect rise of sea-level
- How has Antarctic environment evolved over the years without human interference?



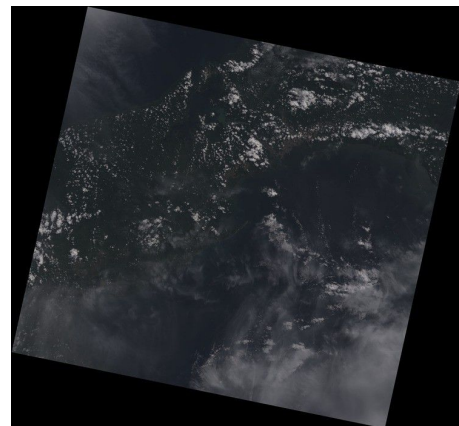
Motivation and Problem Statement (contd.)

- Remote sensing data - huge
- Examining it is time consuming
- Automation of remote sensing analysis - need of the hour to
 - ◆ effectively examine the sensor data and gain valuable insights and trends about the subcontinent
 - ◆ Better information extraction and semantic analysis
 - ◆ Real time analysis of streaming data
 - ◆ Biodiversity tracking, glaciological and other geological studies



What are we doing?

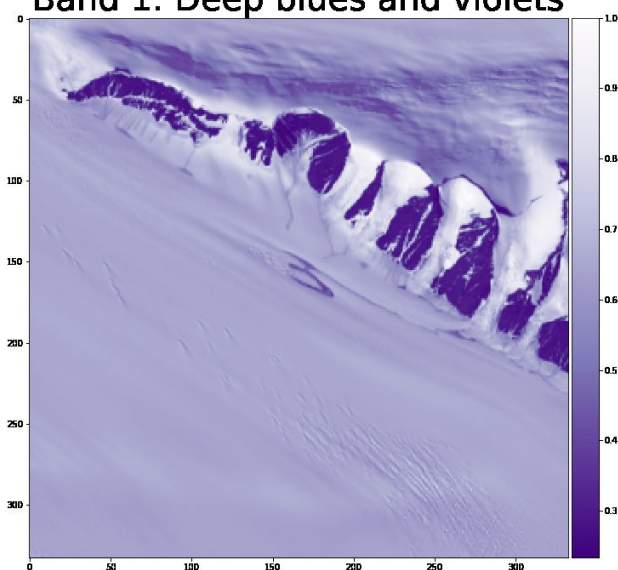
- **Aim** : Automating this process to facilitate research in these domains
- **Focus** : Extract rock outcrop from Antarctic Landsat Imagery using Semantic Segmentation



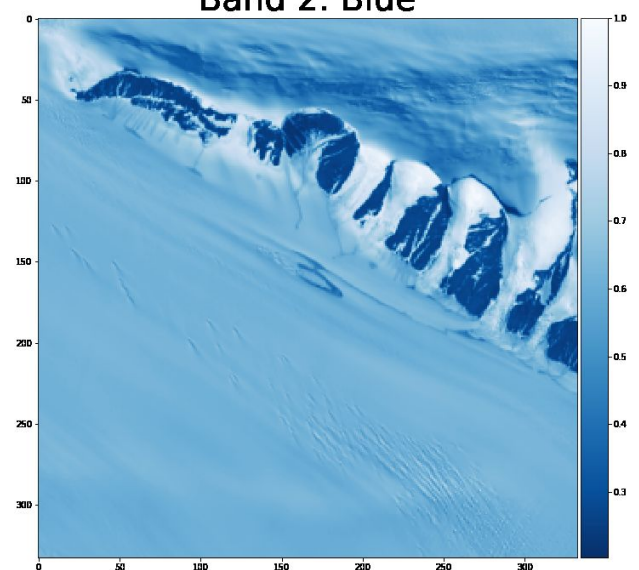
Dataset

- **Dataset** - Landsat 8 Imagery
- **Properties** - Global, Temporal, Hyperspectral
- **Size** - 1 GB per image (approximately)
- **Resolution** - 9000 X 9000 pixels (approximately) 30 meters/pixel
- **Bands:**
 - ◆ **Band 1** senses deep blues and violets.
 - ◆ **Bands 2, 3, and 4** are visible blue, green, and red
 - ◆ **Band 5** measures the near infrared, or NIR
 - ◆ **Bands 6 and 7** cover different slices of the shortwave infrared, or SWIR
 - ◆ **Band 8** is the panchromatic – or just pan – band. It works just like black and white film: instead of collecting visible colors separately, it combines them into one channel.
 - ◆ **Band 9** covers a very thin slice of wavelengths: only 1370 ± 10 nanometers
 - ◆ **Band 10 and 11** are in the thermal infrared, or TIR – they see heat.

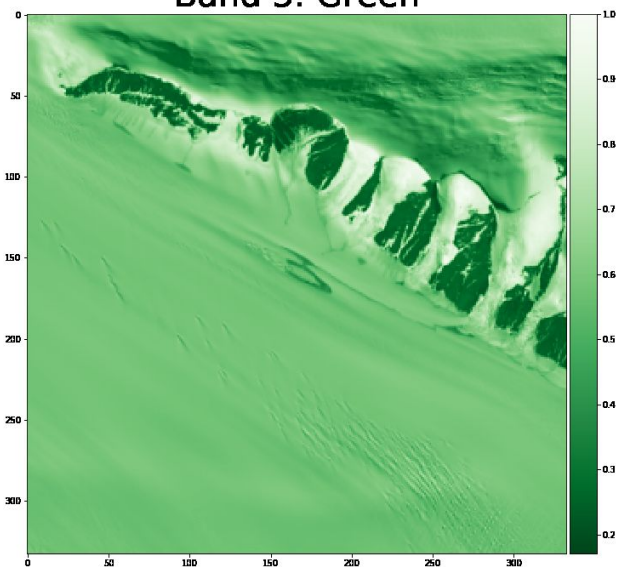
Band 1: Deep blues and violets



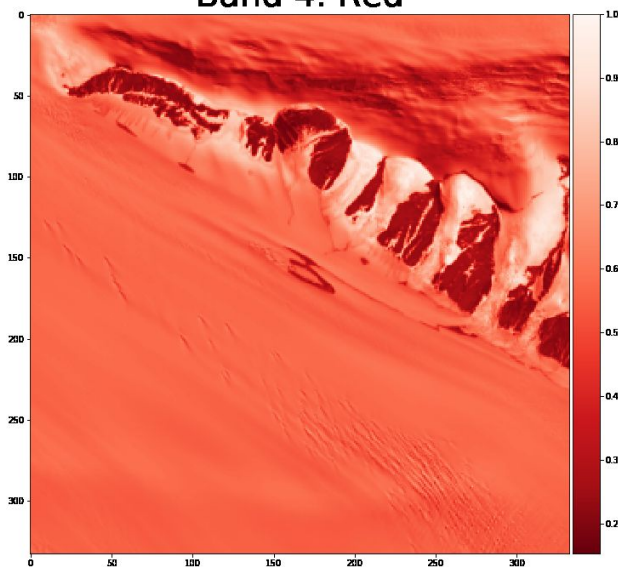
Band 2: Blue



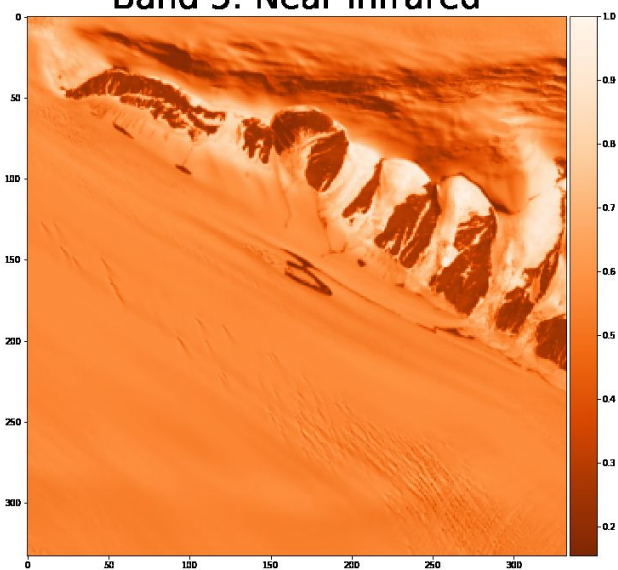
Band 3: Green



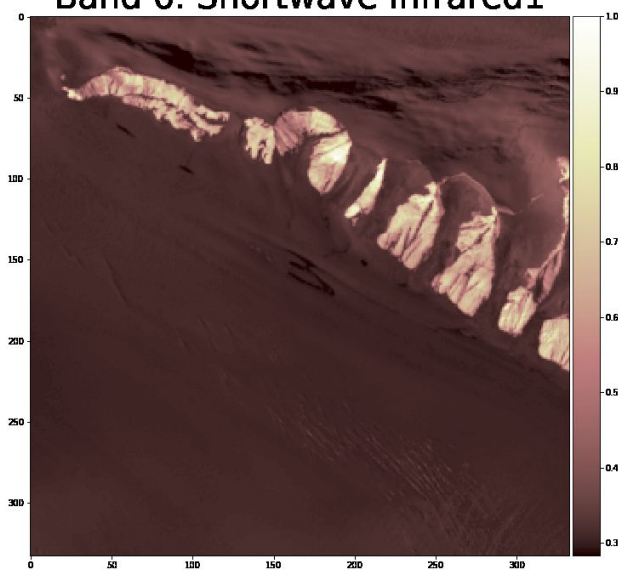
Band 4: Red



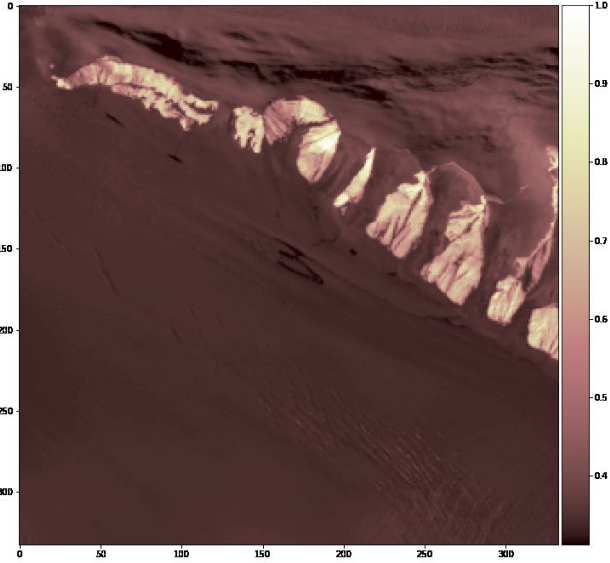
Band 5: Near Infrared



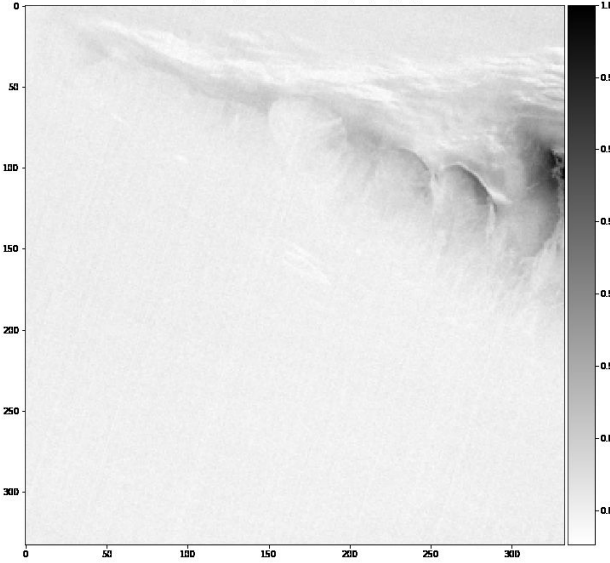
Band 6: Shortwave Infrared1



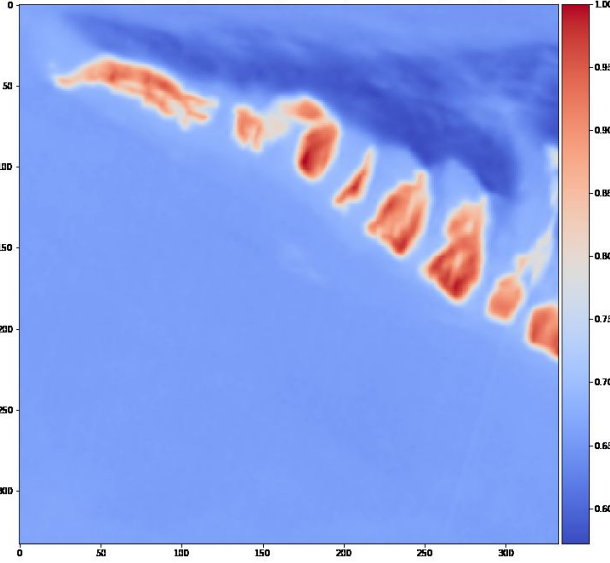
Band 7: Shortwave Infrared2



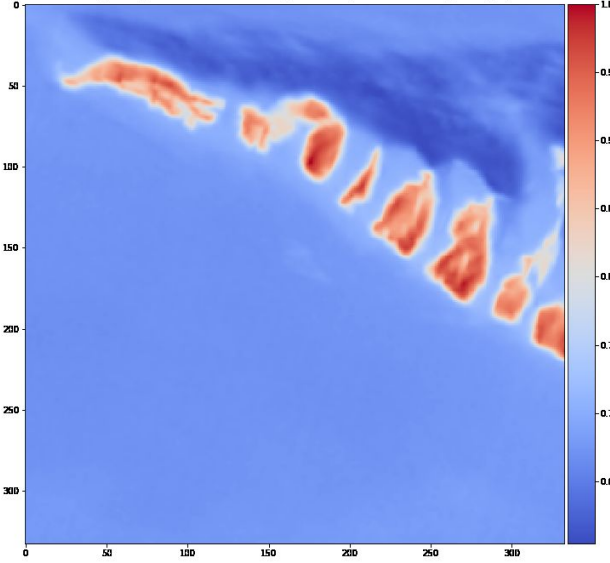
Band 9: Clouds



Band 10: Thermal Infrared



Band 11: Thermal Infrared



Related Work

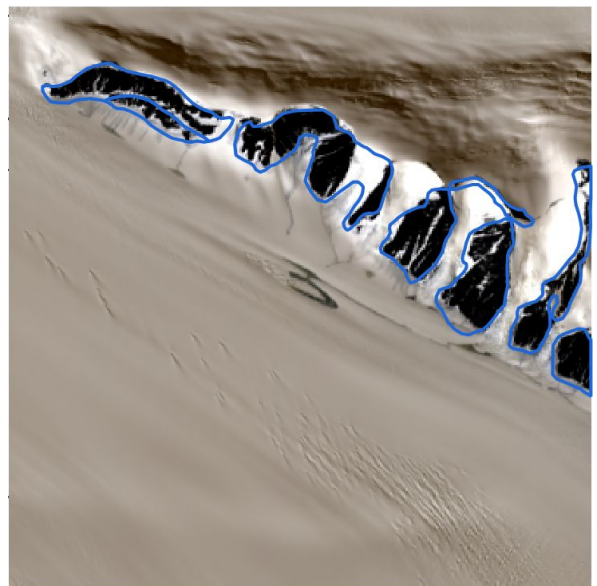
We frame our project as a binary semantic segmentation problem. To approach this problem, we need labels for our dataset and a CNN-based segmentation model to train on the labeled imagery. Manually labeling a sufficient quantity of images to train a model is infeasible.

To choose labels, we researched published continent-scale digital geological datasets. We chose an automatically generated dataset published by Burton-Johnson et. al., 2016. They use pixel-wise classification of Landsat 8 bands to generate a geological dataset. They also publish the list of scenes used as input for their model. Because of this, we can perform a direct comparison of our model to theirs because we have the exact input and output. This enables a more comprehensive understanding of our model's performance.

To choose a model, we found a publication by Chai et. al. in 2019 that used segmentation CNN models to classify clouds, cloud shadows, and ground features in Landsat 8 images. This is very similar to our task and we use the paper to choose a specific model, appropriate training data volume, and hyper parameter starting points.

Related Work

- Several continent-scale geological datasets exist. These include the ADD, the SCAR GeoMap project, and a dataset produced by researchers from the British Antarctic Survey.
- The ADD and GeoMap datasets are manually generated geological maps.
- These are inappropriate as labels because rock features have been generalized to display well on large-scale maps.



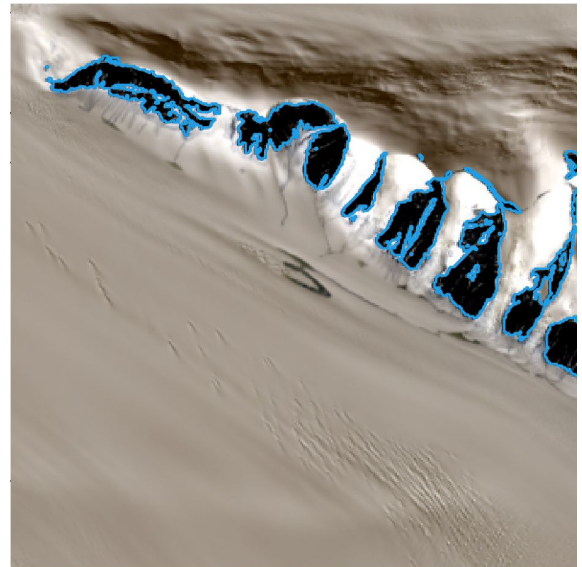
An Example of feature generalization for manually-labeled geological map datasets

© SCAR GeoMAP and GNS Science 2019

Related Work

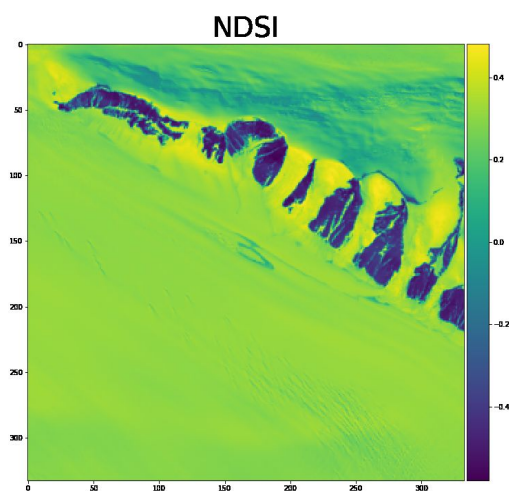
The British Antarctic Survey dataset was generated by an automated methodology that involves masking Landsat 8 images with heuristic thresholds on several band combinations.

This pixel-wise classification yields higher precision and accuracy than any manual method.

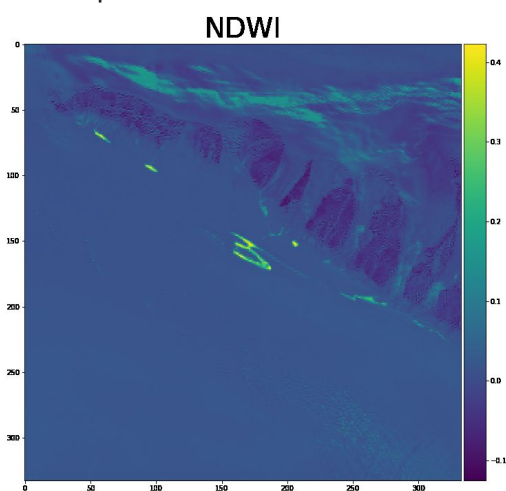


Related Work

Band Combination Examples



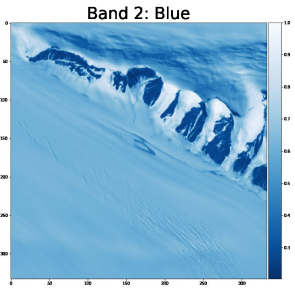
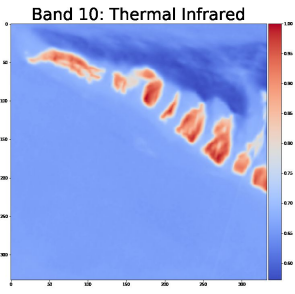
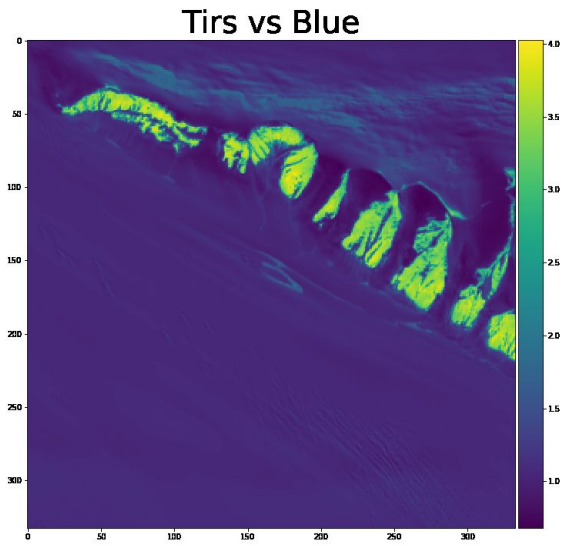
$$\text{NDSI} = \frac{\text{OLI band 3} - \text{OLI band 6}}{\text{OLI band 3} + \text{OLI band 6}}$$



$$\text{NDWI} = \frac{\text{OLI band 3} - \text{OLI band 5}}{\text{OLI band 3} + \text{OLI band 5}}$$

Related Work

Band Combination Examples



Related Work

Heuristic Thresholds

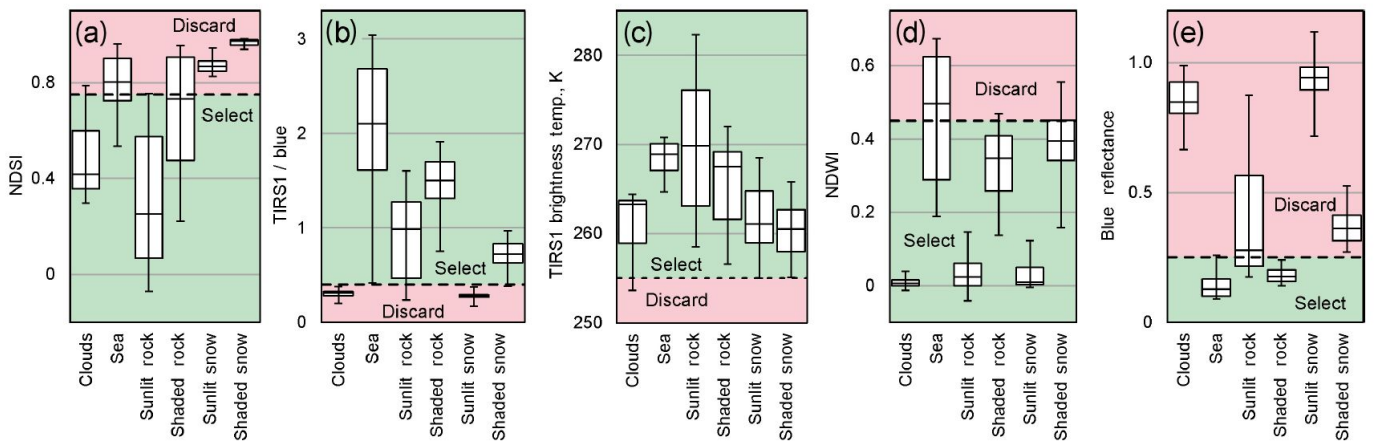
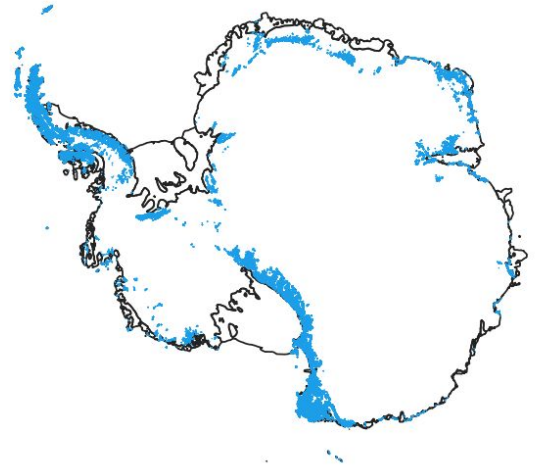


Figure from Burton-Johnson et. al., 2016

Related Work

- Burton-Johnson et. al., 2016
 - ◆ Band combinations used to categorize image features like clouds, sea, sun-lit rock, shaded rock, sun-lit snow, shaded snow.
 - ◆ Use heuristic thresholds on band combinations to isolate sun-lit rock and shaded rock from all other classes of pixels



Results from publication (Table 1.)

Methodology	Correct %	SD	Commission %	SD	Omission %	SD	Classification accuracy %	SD
This study	85	8	17	13	15	8	74	9
Optimum NDSI	68	30	7	6	32	30	63	27
ADD rock outcrop	70	14	154	212	30	14	39	19

Related Work

Chai et. al., 2019 uses PSPNet, Unet and Segnet segmentation models to classify Landsat images into 3 classes: Cloud, cloud shadow, and ground.

Metric	Value
Overall Accuracy	93.45 %
Commission Error	14.36 %
Omission Error	18.98 %

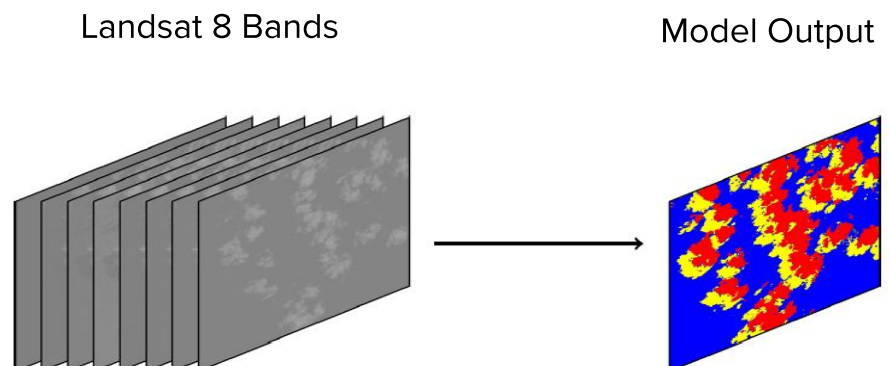


Figure from Chai et. al., 2019

Related Work

- Chai et.al.
 - ◆ Proof that using a semantic segment model is an effective methodology for classifying Landsat 8 imagery.
 - ◆ Data preprocessing methodology accommodates large input file size and allows model to run on “reasonable” amounts of RAM.
 - ◆ Established appropriate data volumes for training an effective classifier.

Table 1
 Training, validation and test sets for L7 Irish and L8 Biome. The number of 512 * 512 30 m images in each set is as follows.

	Images (512 * 512)		
	Train 60%	Val. 10%	Test 30%
L7 Irish	2732	420	1328
L8 Biome	2410	378	1178

Scene Preprocessing

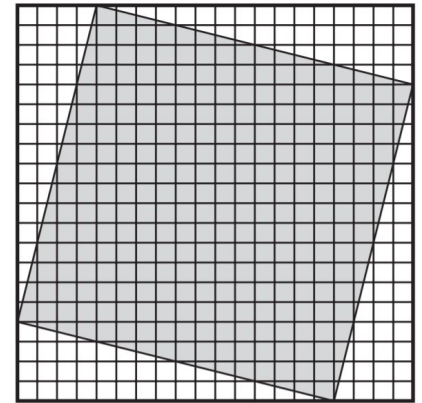


Figure from Chai et. al., 2019

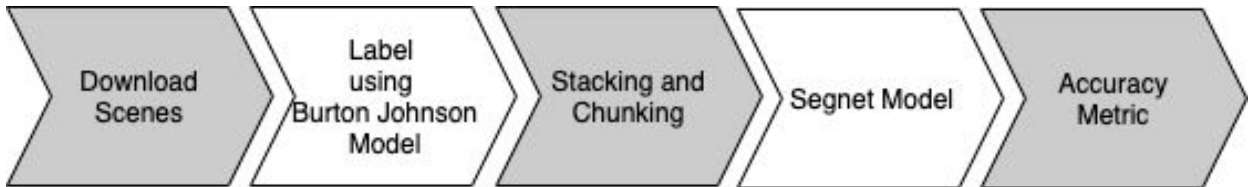
Computing Environment

- Machine Details
 - ◆ Environment- Google Colab
 - ◆ Storage used on Google drive- ~176 GB
 - ◆ GPU specs- Tesla P100 (depending on availability)
 - ◆ Running Time ~ 200 s/epoch (Running time improves on the second run of the same notebook because of caching in Colab)

```

1 import time
2 tic = time.time()
3 print('start time: {}'.format(tic))
4 class_label = 'best_model'
5 model_name = 'best_model'
6 current_model = os.path.join(model_path, model_name)
7 assert os.path.exists(current_model)
8 toc = time.time()
9
10 test_data = read_image_batch(test_image_list, batch_size, test_info["allbands"])
11
12 toc = time.time() - tic
13 print('test_data loaded in {:.2f} seconds\ntotal runtime: {:.2f}'.format(toc, time.time() - tic))
14
15 my_model = create_model()
16
17 toc = time.time() - tic
18 print('model created in {:.2f} seconds\ntotal runtime: {:.2f}'.format(toc, time.time() - tic))
19
20 my_model.compile(optimizer='sgd', loss='categorical_crossentropy')
21 toc = time.time() - tic
22 print('model compiled in {:.2f} seconds\ntotal runtime: {:.2f}'.format(toc, time.time() - tic))
23
24 my_model.load_weights(current_model)
25 toc = time.time() - tic
26 print('weights loaded in {:.2f} seconds\ntotal runtime: {:.2f}'.format(toc, time.time() - tic))
27
28 probabmy_model.predict(test_data, steps=(test_set_size)//batch_size)
29 toc = time.time() - tic
30 print('predictions generated in {:.2f} seconds\ntotal runtime: {:.2f}'.format(toc, time.time() - tic))
31 class_label.append(probe.argmax(axis=-1))
32
33
34 start time: 157472471.0419996
35 test_data loaded in 0.05 seconds
36 total runtime: 0.05
37 model created in 2.03 seconds
38 total runtime: 2.03
39 model compiled in 2.05 seconds
40 total runtime: 4.08
41 weights loaded in 4.13 seconds
42 total runtime: 8.21
  
```

System Architecture



System Architecture Components

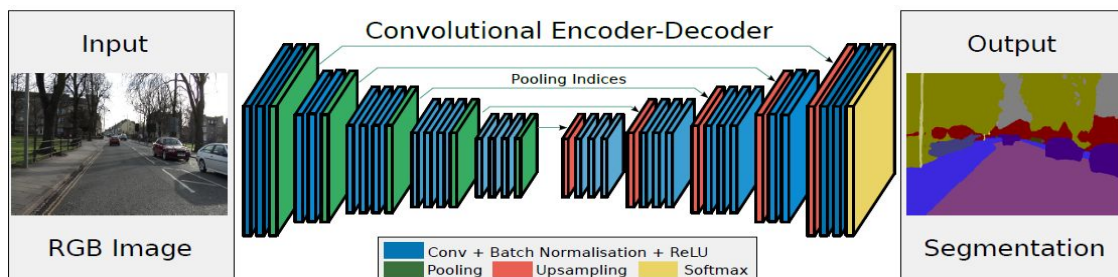
- Download Scene
 - ◆ Script is used to download the scenes from a list of scene ids
 - Label using Burton Johnson
 - ◆ Convert shapefile features that overlap with training scenes to binary raster with 30 m resolution. This serves as a pixel-wise rock/not-rock label.
 - ◆ Labels have intrinsic error from the heuristic thresholds used by Burton-Johnson model
 - Stacking and Chunking
 - ◆ Convert Landsat 8 band .TIF files into a single numpy array (Scene height X Scene width X 11 Landsat bands + 1 label band)
 - ◆ Divide single scene into chunks (512 X 512 X 11 Landsat Bands + 1 label band)
 - Segnet Model
 - ◆ Input: 1 chunk and corresponding label. Output: binary prediction raster
 - Accuracy Assessment
 - ◆ Overlay binary prediction raster with binary ground-truth raster (provided by Alex Burton-Johnson)
 - ◆ Determine True Positives, False Positives, True Negatives, False Negatives.
 - ◆ Calculate accuracy metrics
-

Model Selection- Why Segnet?

- Significantly smaller and faster than other neural architectures
- Small memory and computational requirements, time efficient model
- Helps in delineating the boundaries of objects
- Reduces number of learnt parameters
- Compatible with a wide range of encoding-decoding architectures

System Architecture Components (contd.)

- Segnet
 - ◆ Supervised learning to predict pixel-wise classification labels
 - ◆ It has encoder and decoder layers followed by a pixel-wise classification layer
 - ◆ Encoder units- 13 convolutional layers, 2x2 max-pooling layers
 - ◆ Decoder units- Upsampling, convolutions and per pixel softmax classifier

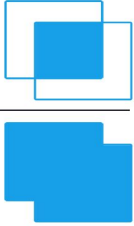


Model Assessment Metrics

→ Accuracy Metrics

- ◆ We didn't choose "Overall Classification Accuracy" as a metric because of the huge class imbalance in our dataset. Since 98% of the pixels are ice, even if our model predicts everything as ice, we get an overall classification accuracy of 98%.
- ◆ So we use accuracy metrics only for our rock predictions. These metrics are - Rock classification accuracy, rock commission error and rock omission error.
- ◆ Rock Classification Accuracy- Pixels that our model predicted correctly as rock. This is the metric that Burton Johnson et. al used to report their accuracy. So we can compare our results to theirs using this metric.

Rock Classification Accuracy =

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


Model Assessment Metrics

→ Accuracy Metrics

- ◆ Rock Commission Error - Pixels that our model predicted as rock but that were not actually rock

$$\text{Rock Commission Error} = \frac{\text{False Positives}}{(\text{False Positives} + \text{True Positives})} = \frac{\text{False Positives}}{\text{Model Output Positives}}$$

- ◆ Rock Omission error - Pixels that our model predicted as "Not Rock" but were actually rock

$$\text{Rock Omission Error} = \frac{\text{False Negatives}}{(\text{False Negatives} + \text{True Positives})} = \frac{\text{False Negatives}}{\text{Ground Truth Positives}}$$

Experiments

- Data : 7000 - 512x512 images
- Test : 9 manually labeled images

Result:

Hyperparameter	Values Tried	Best option
Learning Rate	0.001 to 0.05	0.01
Optimizer	SGD, ADAM	ADAM
Data Volume	1% feature rich data, 5% feature rich data, all data	1% feature rich data
Class Weight	1:1, 1:20, 1:50	1:1

Experiment 1

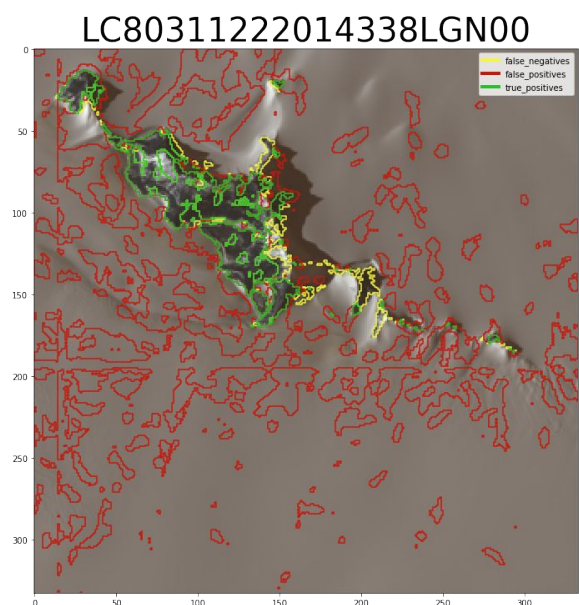
- Learning rate : Multiple experiments with varying learning rates were performed, but we obtained invalid results for all values except 0.01
-

Experiment 2

- Data : Filtered data to use only images which have at least 1% of the pixels labeled as rock.
- Learning rate : 0.01
- Optimizer : SGD
- Classification accuracy : 0.26 ± 0.74 %
- Omission Error : 99.74 ± 0.75 %
- Commission Error : N/A (No rock pixels were incorrectly labeled)

Experiment 3

- Data : Filtered data to use only images which have at least 5% of the pixels labeled as rock.
- Learning rate : 0.01
- Optimizer : Adam
- Classification accuracy : 27.75 ± 21.72 %
- Omission Error : 44.23 ± 25.66 %
- Commission Error : 62.79 ± 26.13 %

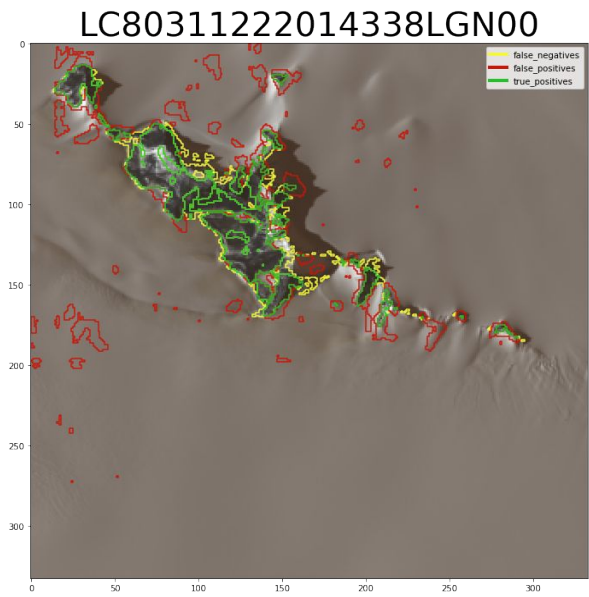


Example of high commission error

Experiment 4 (Best model)

- Data : Filtered data to use only images which have at least 1% of the pixels labeled as rock.
- Learning rate : 0.01
- Optimizer : Adam
- Classification accuracy : 30.48 ± 17.6 %
- Omission Error : 41.05 ± 18.94 %
- Commission Error : 59.57 ± 21.56 %

All further results have been generated from this model.

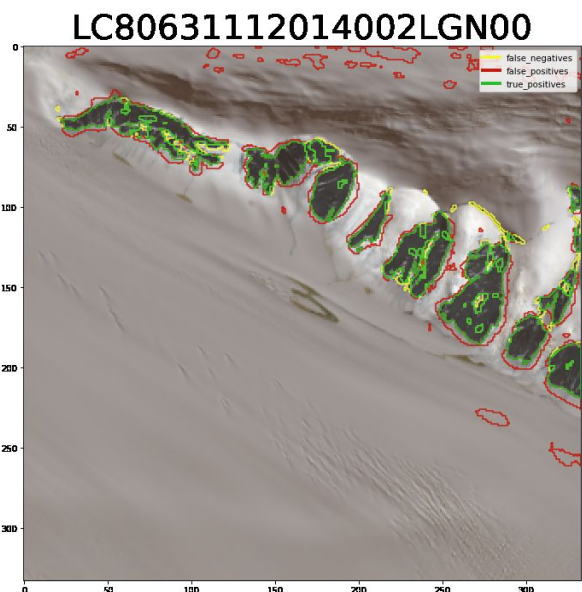


Reduction in commission error

Results

For the presented image

Metric	Value
Classification Accuracy	59.88 %
Commission Error	33.40 %
Omission Error	14.41 %

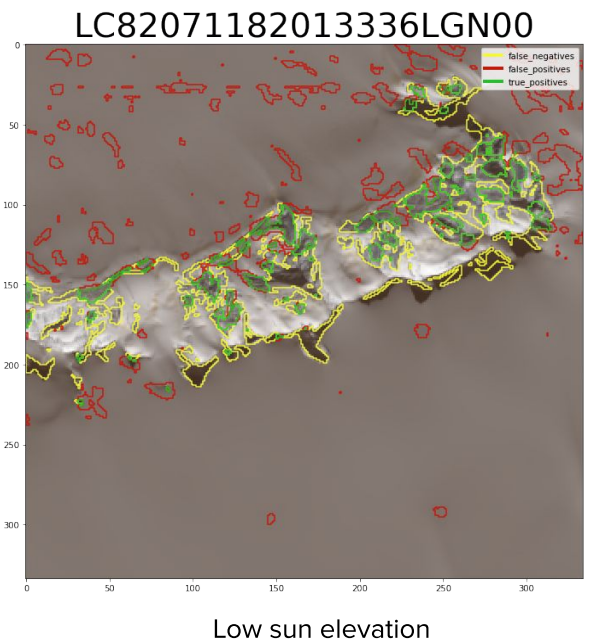


Analysis

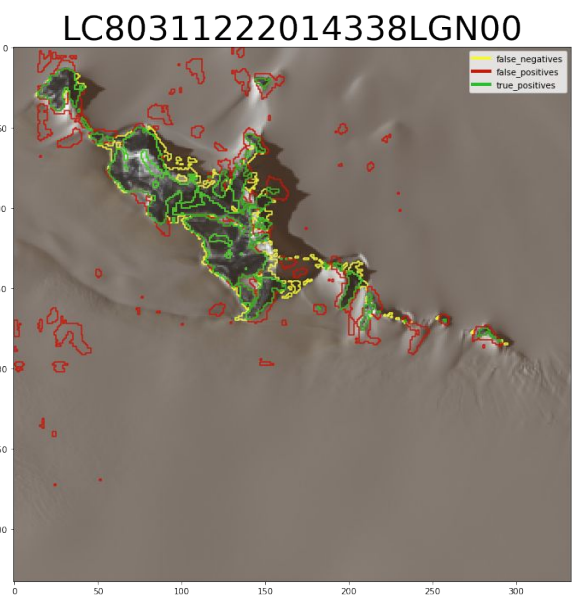
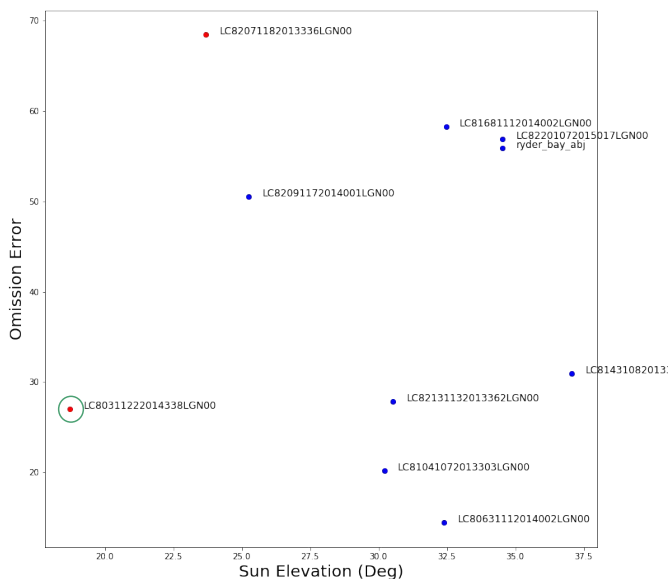
Factors affecting prediction:

- **Sun elevation** - Areas of low sun elevation as that shown in the image, create contrasting regions of sunlit and shaded rock outcrops. We are able to predict most sunlit rocks correctly, but get high omission error in shaded rocks.

The plot in the following slide shows a trend between sun elevation and omission error in all test images. If we consider the circled image as an outlier, we do see a decreasing trend in error as sun elevation increases. The outlier image (shown in next slide) gives low error in spite of low sun elevation because it's a relatively clean image with even textured ice and a single big outcrop of rock.



Analysis (contd.)

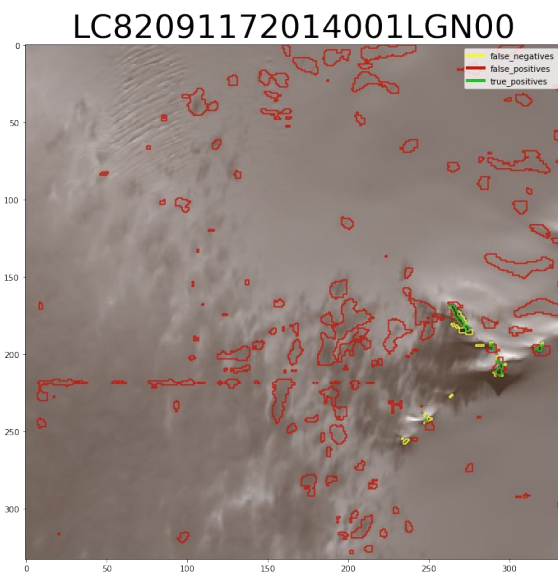


Analysis (contd.)

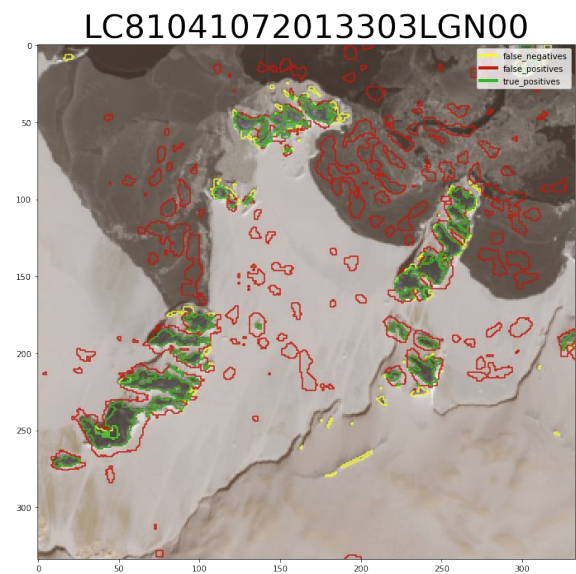
Factors affecting prediction:

- **Cloud cover** - Areas which have dense cloud cover create cloud shadows on the ground which get mispredicted as rocks giving high commission error.
- **Coastal features** - Areas which have a coastline cause high commission error in the regions of melting ice and water.

We observe that plotting these features do not show statistically confident trends. Hence, we stick to a qualitative assessment to see the impact of these factors.



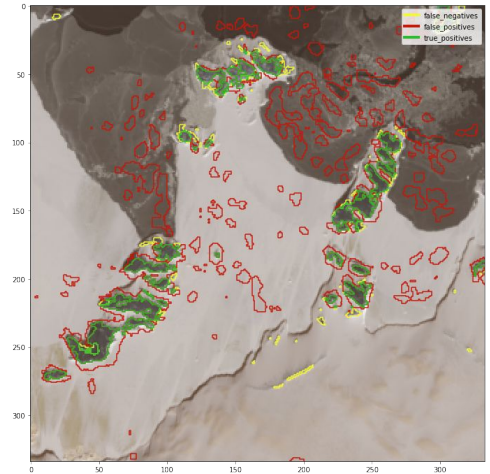
Cloud cover



Coastline

Conclusion

- Observed significant impact of feature density in training data on model performance. 1% feature rich data gave the best results due to the high class imbalance in our dataset
- Established trends in classification accuracy with respect to sun elevation and cloud cover which helped in distinguishing between easy and hard cases
- Our model struggles with same types of hard cases established by Burton Johnson et al - cloud cover and coastline
- Make our code base open source to facilitate future work on this dataset



Conclusion (contd.)

Study	Result (Rock Class Acc)
Our Model	30.48 ± 17.6 %
Burton-Johnson et al., 2016	74 ± 9%
Chai et. al., 2019	93.45% (overall Accuracy)

- Further tuning and training is needed before our model approaches the accuracy of the Burton-Johnson model.
- If the class imbalance problem can be overcome, models could be trained to produce output from spectral bands not considered by the Burton-Johnson model. If the signal from these extra bands are incorporated into an ensemble classifier with other classifiers, the Burton-Johnson accuracy could be exceeded.
- Direct metric comparisons with Chai et. al. are not possible, but further tuning and training is needed before concluding whether or not CNN-based segmentation models are appropriate for this classification task.

Future Work

- Ensembling with different band combinations or different input subsets or different CNN models
- Instead of binary classification - classification into multiple classes - ice, rocks, clouds, water
- Use noise correction techniques to reduce effect of noise in training data. We can use more deconvolution layers to denoise images. We can also use classic computer vision techniques like Median Blurring and Gaussian blurring before training
- Use different models like UNet or PSPNet instead of Segnet
- Use transfer learning on pretrained models
- Use semi-supervised learning instead of using Burton-Johnson labels - We can cluster similar data using unsupervised learning and use the 9 manually labelled scenes to label the data
- Refine heuristic thresholds to improve label quality

References

- [1] Alex Burton-Johnson, Martin Black, Peter Fretwell, and Joseph Kaluza-Gilbert. An automated methodology for differentiating rock from snow, clouds and sea in antarctica from landsat 8 imagery: a new rock outcrop map and area estimation for the entire antarctic continent. *The Cryosphere*, 10:1665–1677,2016.
- [2] Martin Långkvist, Andrey Kiselev, Marjan Alirezaie, and Amy Loutfi. Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8(4):329, 2016.
- [3] Xuemei Zhao, Lianru Gao, Zhengchao Chen, Bing Zhang, and Wenzhi Liao. Cnn-based large scale landsat image classification. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPAASC)*, pages 611–617. IEEE, 2018.
- [4] Dengfeng Chai, Shawn Newsam, Hankui K Zhang, Yifan Qiu, and Jingfeng Huang. Cloud and cloud shadow detection in landsat imagery based on deep convolutional neural networks. *Remote sensing of environment*, 225:307–316, 2019.
- [5] Alex Kendall, Vijay Badrinarayanan, , and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.
- [6] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [7] Cox S.C., Smith Lyttle B. and the GeoMAP team (2019). Lower Hutt, New Zealand. GNS Science. Release v.201907.
-

Thank You!

