

Project outside course scope

Supervised Land-Surface Segmentation

Satellite Image Analysis and Modelling

Author: Johannes Brøns Christensen (swd122)

Supervisor: François Bernard Lauze

Date of Submission: 21th of August, 2023

Abstract

Accurately modeling ice sheet movement in Greenland is crucial for comprehending climate change impacts. This study introduces innovative satellite image segmentation techniques using U-net architectures to precisely delineate land regions along Greenland's coast. Focusing on land segmentation complements existing methods for ice and water segmentation, augmenting comprehensive ice sheet movement models. The proposed methodology is validated on satellite images, including islands, utilizing standard evaluation metrics. The outcomes provide valuable insights for climate experts, enhancing the precision of ice sheet models and enriching our understanding of climate change effects on polar regions.

Keywords: Satellite Segmentation, U-net, Machine Learning, Greenland, Climate Crisis

Contents

1	Introduction	5
2	Data	5
2.1	Availability	5
2.2	Assumptions and Relevant Subset	6
2.3	Cloud Coverage	6
2.3.1	Spatial Resolution	6
2.4	Bands	6
2.5	Areas of Interest (AOIs)	6
2.6	File Encoding	7
2.6.1	GeoTIFF	7
2.6.2	Shapefile	9
2.6.3	Pre-processing	10
2.7	Merging Tiles	11
3	Methods	12
3.1	Semantic Segmentation Model	12
3.2	Performance Metrics	13
3.2.1	Class Imbalance and Pixel Accuracy	13
3.2.2	Precision	14
3.2.3	Recall	14
3.2.4	Intersection over Union (IoU)	14
3.2.5	Dice Score	14
3.3	Dice Loss	15
3.4	U-Net Architecture for Semantic Segmentation	15
3.5	Transfer Learning	17
3.6	GPU Cluster Efficiency	17
4	Experiments and Results	18
4.1	Mosaic Creation	18
4.2	Benchmarking	19
5	Discussions and Conclusions	19
5.1	Overall Performance	19
5.2	Mask Quality and Limitations	20
5.2.1	Resolution and Human Error	20
5.2.2	Expected Outcome and Proposed Solution	20
5.3	Limitations	21
5.3.1	Mosaic Creation	21

5.3.2	Data and Model Training	21
5.4	Proposed Methods for Future Adaptation	22
5.4.1	Active Learning	22
5.4.2	Online Learning	22
A	Appendix	26
A.1	Planet Order	26
A.2	Different mosaic creation with mask	28
A.3	Island Mask Zoomed	29
A.4	Project Description of the Contract	29

1 Introduction

The escalating climate crisis has spurred the need for monitoring Greenland’s changing ice sheet mass and its consequential effects on land surfaces and sea levels[20]. Notably, a larger-scale project called ”An unexpectedly large count of trees in the West African Sahara and Sahel” [1] has already explored advanced techniques for dealing with Earth’s curvature and employed active learning strategies, showcasing the potential of online learning.

Manual inspection and deterministic methods currently underpin Greenland’s coastline monitoring, posing challenges in scalability and practicality. This study proposes innovative U-net architectures for satellite image segmentation, automating land region delineation. This approach streamlines monitoring, enabling focused analysis of evolving areas.

The presented methodology for land segmentation is initially trained, validated, and tested on a limited dataset of 4 separate images. Successful implementation holds potential to enhance tracking Greenland’s ice sheet dynamics’ accuracy and efficiency. While this implementation provides a foundational pipeline, a more comprehensive approach would encompass online and active learning. Despite this, the initial pipeline contributes to a deeper understanding of climate change impacts on polar regions and global sea levels.

2 Data

For this project, high resolution 4-band satellite image data is obtained through the proprietary service known as *Planet*[12, 13]. The provider is what tenure track adjunkt geographer, Anders Anker Bjørk[3], uses for his research in the changing water levels across the Greenlandic coastline[8, 21, 22].

2.1 Availability

The data coming from the Planet service is proprietary . Nonetheless, one might be granted limited access via account creation[12]. The target data was provided for the project by Anders Anker Bjørk[3] formatted as *shape-files*.

The source code of this project [2] is available on GitHub under the MIT licence at:

<https://github.com/JohannesBroens/Automatic-Satellite-Island-Discovery>.

2.2 Assumptions and Relevant Subset

2.3 Cloud Coverage

This project exclusively employs satellite images from Planet that have been certified to possess negligible cloud cover[15]. Cloud formations significantly disrupt image integrity, impeding the model’s capacity to learn accurate representations.

2.3.1 Spatial Resolution

The spatial resolution of a satellite image, quantified in square meters per pixel. Notably, lower resolution images entail reduced hardware resource costs. Given limited resources, obtaining and processing larger, high-resolution areas may prove infeasible. However, high spatial resolution remains imperative for precise island segmentation. Consequently, this project selects a maximal resolution of approximately $3 \frac{\text{m}^2}{\text{pixel}}$, as outlined by PlanetScope data.

2.4 Bands

The model harnesses four spectral channels, often termed as bands: red, green, blue, and near-infrared (NIR) [14]. Of these, the NIR channel provides the model with the most informative features concerning the target areas. Extending the channel count to eight is not expected to furnish a clear boost in information, considering the additional memory bandwidth required.

2.5 Areas of Interest (AOIs)

Although the methods developed in this study can be applied to arctic regions in general, the project’s focus is on four specific areas of interest along the northwestern coastline of Greenland. This selection is designed to optimize resources, allocating two areas for training, one for validation, and one for testing.

Temporal considerations constrain the project to the "late summer" time-frame, ensuring sufficient sunlight and minimal expected ice coverage in the AOIs[10, 11]. Additionally, the projected rise in annual temperatures is anticipated to reduce ice coverage in the coming years. Consequently, the dataset consists of satellite images captured during the late summer of the preceding year, 2022.

2.6 File Encoding

2.6.1 GeoTIFF

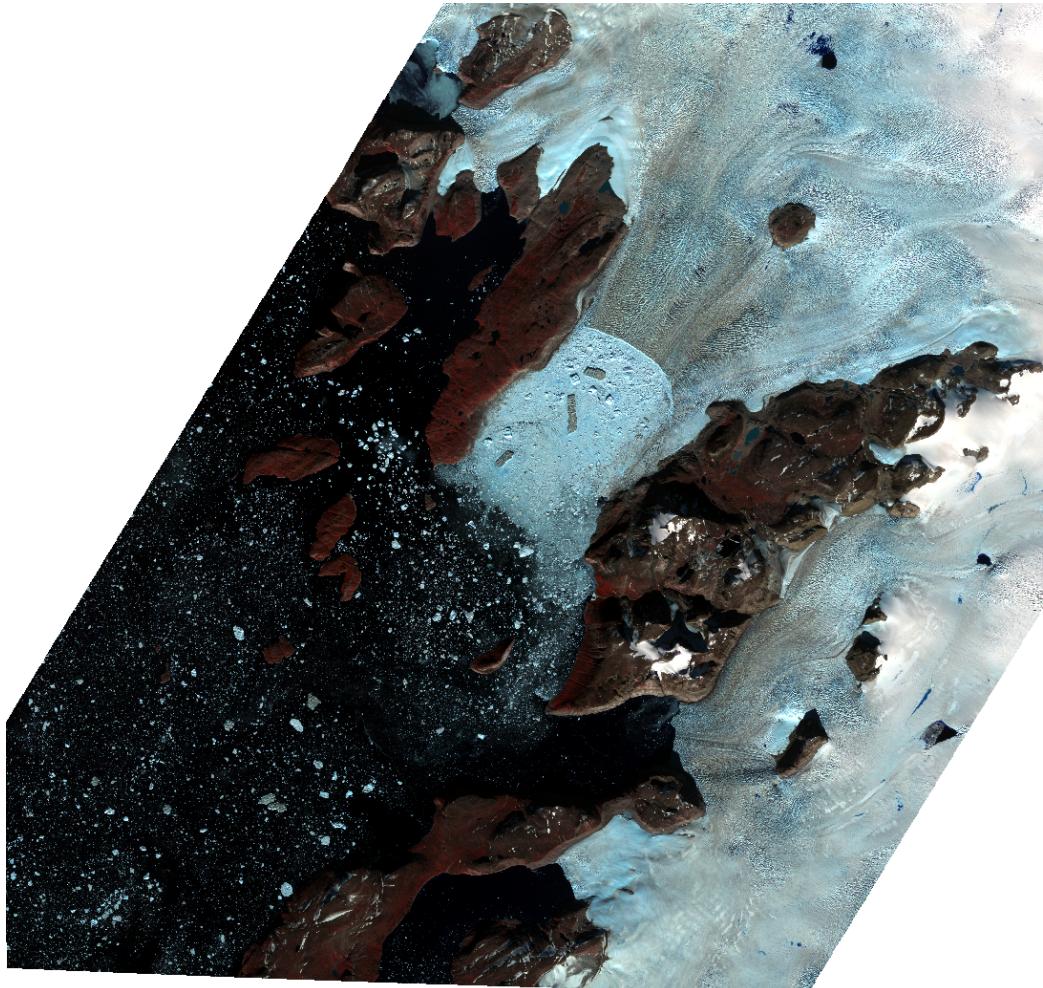


Figure 1: First satellite image. 512×512 segments of this image is used for training the U-Net.

Satellite imagery contains multi-spectral pixel-intensity values, offering valuable scene details. Georeferencing metadata, tied to each pixel, imparts crucial geographical context. Converting pixel coordinates to geographical coordinates involves intricate transformations that align the imagery with real-world locations [17]. These transformations map pixel positions to a chosen Coordinate Reference System (CRS).

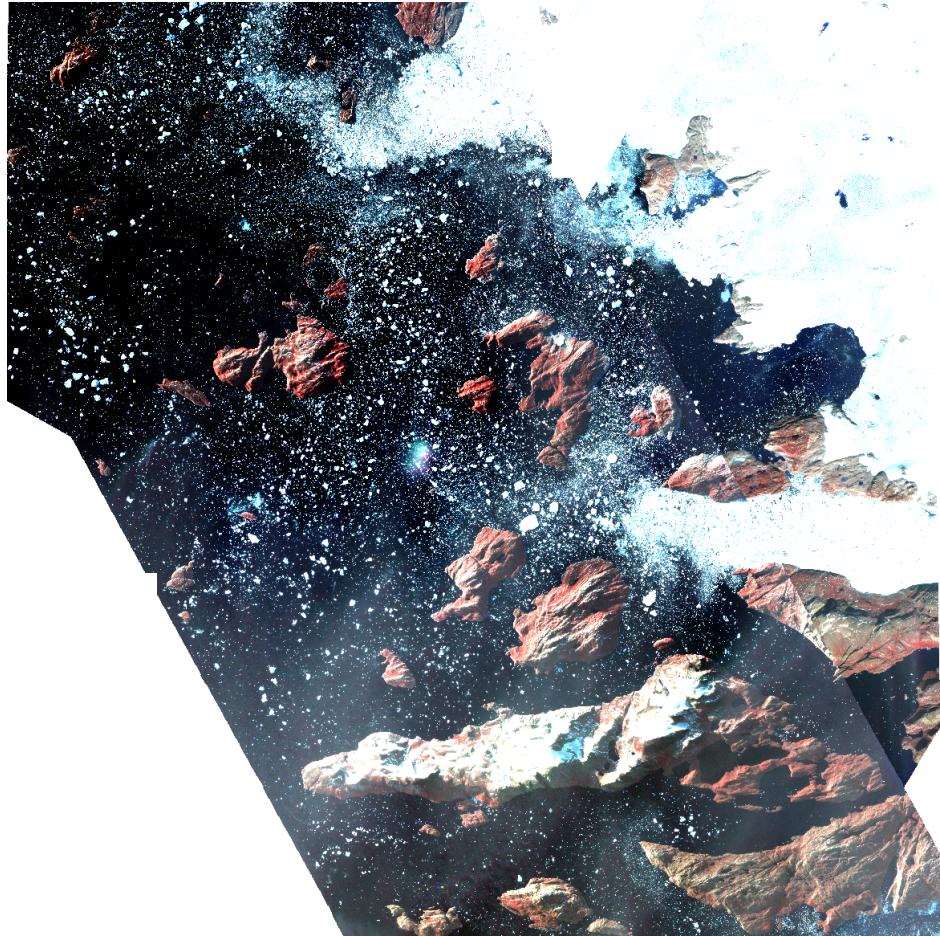


Figure 2: Second satellite image. 512×512 segments of this image is used for training.

A common transformation leverages a 2×2 rotation matrix \mathbf{R} and a 2×1 offset vector \mathbf{O} . Given pixel coordinates $\mathbf{X}_{\text{pixel}} = [x, y]$, the transformation to geographical coordinates $\mathbf{X}_{\text{geo}} = [X, Y]$ is:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \mathbf{R} \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{O}.$$

While 2×2 transformations and offsets align imagery well on locally small areas, the curvature of the Earth introduces complexities when larger areas are transformed this way, as illustrated in Figure 5. Larger images may exhibit localized distortions due to the curved surface. Sophisticated map projections and transformations address such curvature effects.

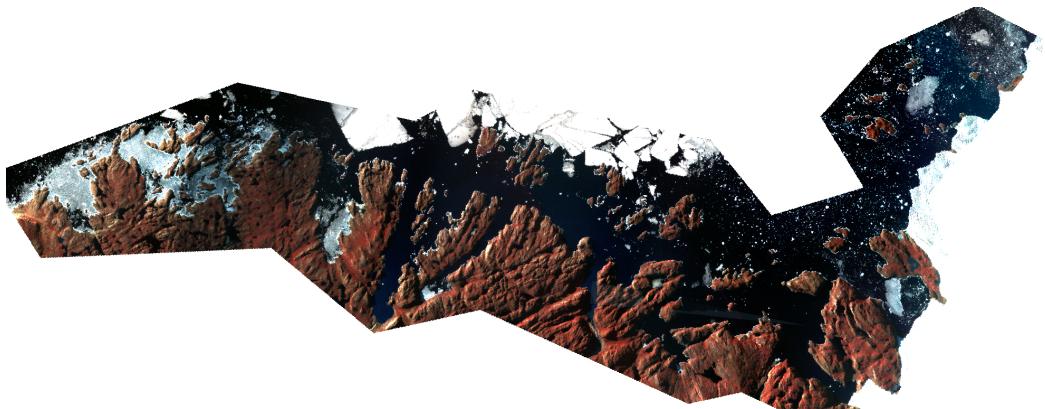


Figure 3: Third satellite image. 512×512 segments of this image is used for validating the model.

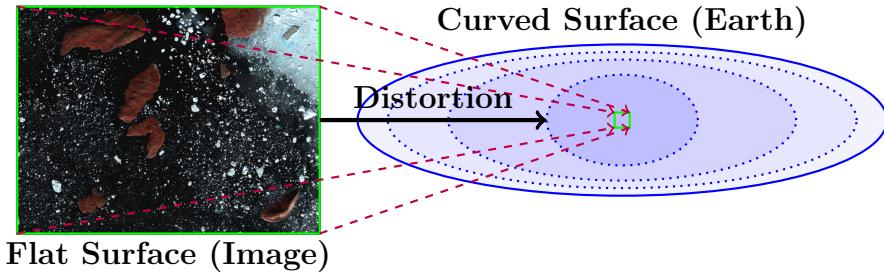


Figure 5: Illustration of Distortion due to Earth’s Curvature. Larger images may exhibit localized distortions on the curved Earth’s surface, as shown by the connecting arrows. The red dashed contour indicates the curvature.

Various transformations, beyond 2×2 matrices and offsets, handle rotation, scaling, and skew. These ensure more precise mapping on the curved Earth’s surface.

GeoTIFF files, denoted as **.tif**, store pixel data and geospatial metadata, accurately positioning images on Earth’s curved terrain. Geographical coordinate references can be removed during pre-processing, retaining pixel information.

2.6.2 Shapefile

Segmentation masks are meticulously crafted by geographers in Geographic Information Systems (GIS). This meticulous process generates geospatial *vector files*, commonly referred to as *shapefiles*[6], housing multiple polygons.

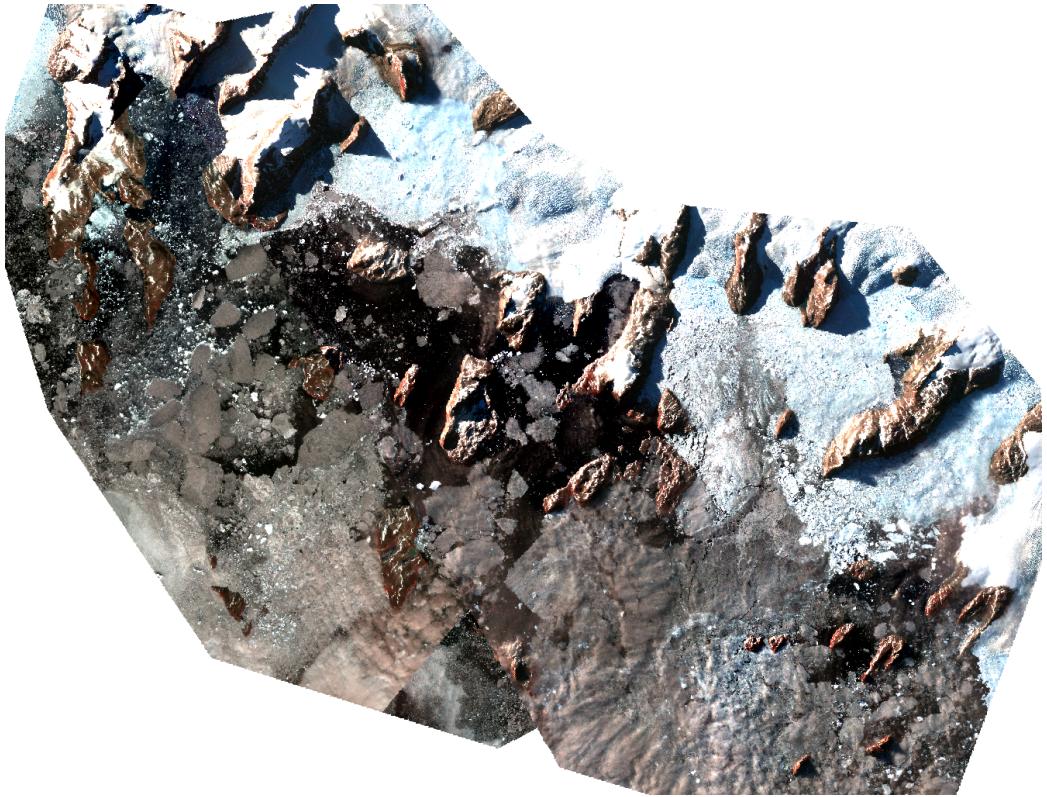


Figure 4: Fourth satellite image. 512×512 segments of this image is used for testing the performance of the final U-Net model.

One merit of this format is its adaptability to diverse spatial resolution satellite images. The project employs shapefiles of varying qualities. The island shapefile originates from high-quality satellite images between 2015 and 2020. Conversely, the remaining shapefiles are of lower quality, sourced from unknown origins. Further limitations of these masks will be discussed subsequently.

2.6.3 Pre-processing

Shapefiles are transformed into **GeoTiffs**, aligned with the Area of Interest (AOI) and spatial resolution of satellite images. Both mask and satellite image are converted to *PyTorch* tensors[9] for efficient training. The mask is binary, with each pixel as 0 or 1, denoting land presence.

The target mask combines three shapefiles: islands, mainland (largely ice-covered), and the ice sheet. Note that only the island shapefile is high-

quality, while the mainland and ice sheet shapefiles are of poor quality.

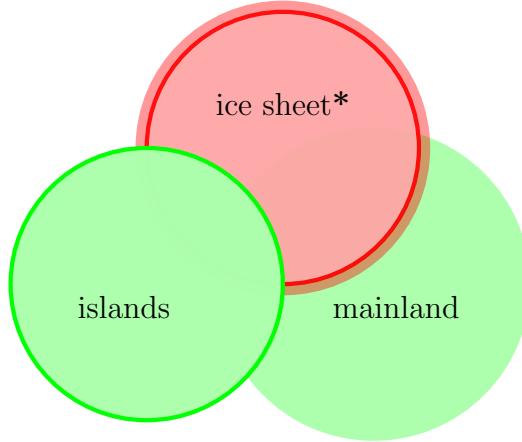


Figure 6: The green color illustrates what is kept as target mask.

As illustrated in figure 6, the target mask is defined as:

$$Y_{\text{target}} = \text{islands} \vee (\text{mainland} \wedge \neg(\text{ice sheet}^*))$$

Here, \vee corresponds to set union, combining islands with mainland or ice-free regions. \wedge is set intersection, including mainland regions not part of the ice sheet. \neg is set complement, excluding ice sheet areas.

(*) dilation expands ice sheet regions, addressing floating-point errors, fixing boundary ambiguities by extending ice sheet boundaries.

Islands alone as the target mask could hinder learning due to similar coastlines. Objective is not classifying ice sheets or snow as land. Hence, ice-covered mainland areas are excluded from interest.

2.7 Merging Tiles

Satellite images are composed of smaller overlapping images, known as *tiles*, which represent different sections of the same scene. These tiles are captured within a short time-frame. Merging these tiles is essential to create a comprehensive view, referred to as a *mosaic* or a *virtual raster image* [18].

Tile merging takes into account the Earth’s curvature to ensure accurate georegistration. Alignment, radiometric adjustments, and spatial coherence are managed carefully. Aggregation strategies involve combining pixel values from overlapping areas of adjacent tiles to achieve smooth transitions.

For this project, the chosen method is the *nearest-neighbor* strategy, where mosaic pixel values come directly from the nearest tile pixels. How-

ever, this simplicity might result in visible discontinuities or artifacts. Alternatively, *bilinear interpolation* computes weighted averages of surrounding pixels, yielding smoother but blurrier transitions.

Aggregation methods are adaptable based on desired outcomes and image characteristics. Advanced techniques, such as radiometric corrections, resampling, and filtering, enhance mosaic quality while considering the Earth’s curvature.

The resultant mosaic serves as the foundation for subsequent analyses, including image segmentation. Preserving spatial integrity is crucial, as inaccuracies introduced during merging can significantly affect downstream tasks.

3 Methods

3.1 Semantic Segmentation Model

Let \mathbf{I} be the input satellite image represented as a tensor of shape (C, H, W) , where C denotes the number of channels (e.g. color channels) and H and W represent the height and width of the image, respectively. Each element $\mathbf{I}_{c,i,j}$ corresponds to the feature value in channel c at pixel location (i, j) .

The objective of semantic segmentation is to assign a class label to each pixel, indicating the object or region it belongs to.

A semantic segmentation model \mathcal{F}_θ can be formulated as a function parameterized by θ , taking the input image tensor \mathbf{I} and generating an output prediction tensor \mathbf{P} with the same spatial dimensions:

$$\mathbf{P} = \mathcal{F}_\theta(\mathbf{I}).$$

The prediction tensor \mathbf{P} has a shape of (H, W) for binary segmentation. For each location (i, j) , the prediction tensor provides a probability $p_{i,j}$ that the pixel belongs to the target class.

The final predicted binary label for each pixel is determined by thresholding the predicted probabilities:

$$y_{i,j} = \begin{cases} 1, & \text{if } p_{i,j} > \text{threshold}, \\ 0, & \text{otherwise.} \end{cases}$$

During training, the model’s parameters θ are learned by minimizing a suitable loss function \mathcal{L} that quantifies the discrepancy between the predicted probabilities \mathbf{P} and the ground truth binary mask \mathbf{Y} :

$$\mathcal{L}(\mathbf{P}, \mathbf{Y}) = \sum_{i,j} \ell(p_{i,j}, y_{i,j}),$$

where ℓ is an element-wise loss function, often chosen as the binary cross-entropy loss.

A popular architecture for semantic segmentation tasks is the U-Net [16]. As described in 3.4, the U-Net architecture consists of a contracting path that captures context and a symmetric expanding path for precise localization. U-Net's skip connections facilitate information transfer between the contracting and expanding paths, aiding segmentation accuracy.

Gradient descent algorithms, like stochastic gradient descent (SGD), are employed to optimize the segmentation model's (neural network) parameters θ by iteratively adjusting them to minimize the loss \mathcal{L} . SGD computes gradients of the loss with respect to the parameters and updates them accordingly.

3.2 Performance Metrics

To assess the effectiveness of the semantic segmentation model, various performance metrics are used to evaluate the model's predictions against the ground truth.

Notation Explainer

H and W : Height and width of the image.

$y_{i,j}$: True label at pixel location (i, j) .

$\hat{y}_{i,j}$: Predicted label at pixel location (i, j) .

$p_{i,j}$: Logits-predictions by the model at pixel location (i, j) .

$I(\text{condition})$: Indicator function; 1 if condition, 0 otherwise.

ϵ : Small real positive constant to avoid zero division.

3.2.1 Class Imbalance and Pixel Accuracy

Pixel accuracy calculates the ratio of correctly classified pixels to the total number of pixels in an image. However, in binary segmentation tasks with imbalanced classes, this metric can be misleading due to the dominance of one class. For instance, if most pixels are from the negative class, a high accuracy can result from classifying everything as negative. The pixel accuracy is calculated as:

$$\text{Pixel Accuracy} = \frac{\sum_{i,j} I(y_{i,j} = \hat{y}_{i,j})}{H \times W}$$

3.2.2 Precision

Precision quantifies the accuracy of positive predictions made by the model. It is defined as the ratio of true positive predictions to the total number of positive predictions. Precision is a useful metric when false positives need to be minimized.

$$\text{Precision} = \frac{\sum_{i,j} I(y_{i,j} = 1 \wedge \hat{y}_{i,j} = 1)}{\sum_{i,j} I(\hat{y}_{i,j} = 1)}$$

3.2.3 Recall

Recall, also known as sensitivity or true positive rate, measures the model's ability to correctly identify positive instances. It is calculated as the ratio of true positive predictions to the total number of actual positive instances. Recall is particularly important when false negatives should be minimized.

$$\text{Recall} = \frac{\sum_{i,j} I(y_{i,j} = 1 \wedge \hat{y}_{i,j} = 1)}{\sum_{i,j} I(y_{i,j} = 1)}$$

3.2.4 Intersection over Union (IoU)

The Intersection over Union (IoU), also known as the Jaccard index, assesses the overlap between the predicted segmentation and the ground truth. It is defined as the ratio of the intersection of the predicted and true positive areas to the union of these areas. IoU ranges between 0 and 1, where higher values indicate better segmentation accuracy.

$$\text{IoU} = \frac{\sum_{i,j} I(y_{i,j} = 1 \wedge \hat{y}_{i,j} = 1)}{\sum_{i,j} I(y_{i,j} = 1 \vee \hat{y}_{i,j} = 1)}$$

3.2.5 Dice Score

The Dice score, also known as the F_1 -score, evaluates the similarity between the predicted segmentation and the ground truth. It is calculated as twice the intersection of the predicted and true positive areas, divided by the sum of their areas. Like the IoU, the Dice score also ranges between 0 and 1, with higher values indicating better performance.

$$\text{Dice Score} = \frac{2 \sum_{i,j} I(y_{i,j} = 1 \wedge \hat{y}_{i,j} = 1)}{\sum_{i,j} I(y_{i,j} = 1) + \sum_{i,j} I(\hat{y}_{i,j} = 1)}$$

3.3 Dice Loss

The Dice loss is a widely used loss function in semantic segmentation tasks due to its ability to handle imbalanced classes and capture fuzzy boundaries effectively. It is derived from the Dice coefficient, which quantifies the similarity between two sets.

Given a set of predicted values \mathbf{P} and the corresponding target values \mathbf{Y} , the Dice loss $\mathcal{L}_{\text{Dice}}$ is calculated as follows:

$$\mathcal{L}_{\text{Dice}}(\mathbf{P}, \mathbf{Y}) = 1 - \frac{2 \cdot \sum_{i,j} [p_{i,j} \cdot y_{i,j}] + \epsilon}{\sum_{i,j} [p_{i,j}^2 + y_{i,j}^2] + \epsilon}.$$

In this formulation, $p_{i,j}$ and $y_{i,j}$ are the predicted and target element values on the i -th row and j -th column, respectively. The Dice loss captures the overlap and agreement between the predicted and target masks, yielding a value between 0 and 1, where 0 indicates complete mismatch and 1 indicates perfect alignment.

The Dice loss focuses on the structural similarities between the predictions and targets, making it particularly well-suited for tasks involving segmentation of objects and regions in satellite imagery[19]. While the Dice loss does not incorporate the Binary Cross-Entropy (BCE) component present in DiceBCE, its ability to emphasize object boundaries and manage class imbalance can contribute to accurate and meaningful segmentation results.

3.4 U-Net Architecture for Semantic Segmentation

The U-Net architecture is a convolutional neural network (CNN) design tailored for high-resolution image segmentation tasks. It excels in capturing complex spatial relationships and retaining contextual information, making it well-suited for segmenting objects and regions within satellite imagery.

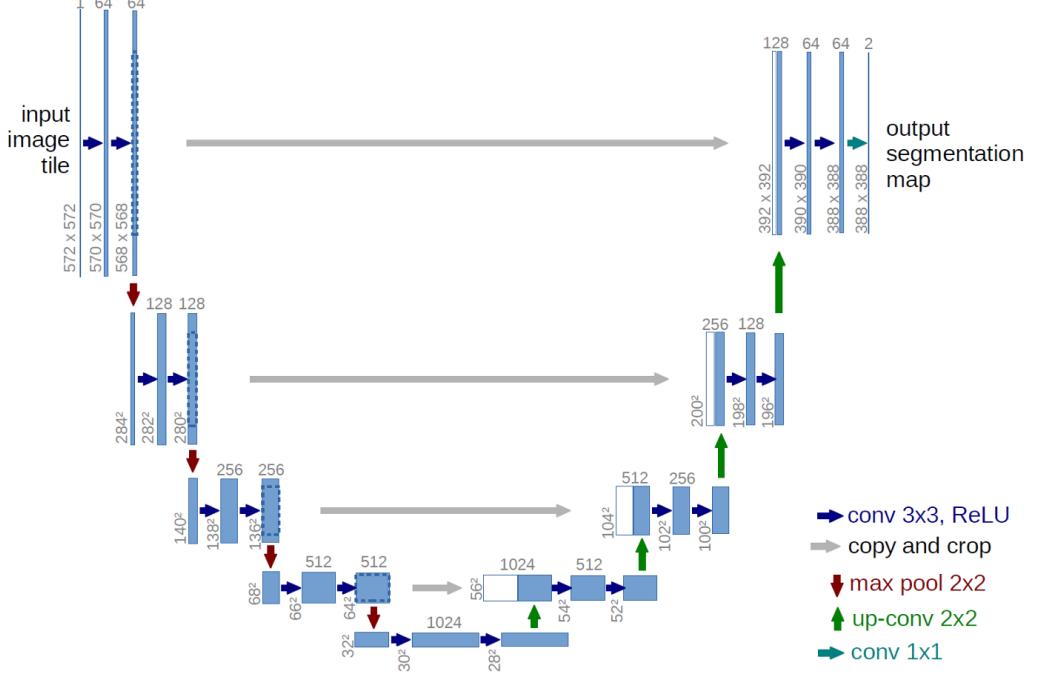


Figure 7: Illustration of U-Net from the original paper by Ronneberger, Fischer, and Brox [16].

The U-Net’s structure can be described as follows. Let \mathbf{X} denote the input image, and \mathbf{Y} represent the corresponding segmented output. The encoding path applies a series of convolutions and max-pooling operations:

$$\mathbf{F}_i = \text{Conv}(\mathbf{F}_{i-1}), \quad \mathbf{F}_i = \text{MaxPool}(\mathbf{F}_{i-1}),$$

where \mathbf{F}_i represents the feature maps at level i . The decoding path uses transposed convolutions for upsampling and fusing information:

$$\mathbf{F}_{i-1} = \text{ConvTranspose}(\mathbf{F}_i) + \mathbf{F}_{i-1},$$

where $\text{ConvTranspose}(\cdot)$ is the transposed convolution operation.

The crucial skip connections in the U-Net are expressed as:

$$\mathbf{U}_i = \mathbf{F}_i \oplus \mathbf{F}_{\text{skip},i},$$

where \oplus represents the element-wise addition, and $\mathbf{F}_{\text{skip},i}$ corresponds to the feature maps from the encoding path at the same level i . This fusion effectively combines high-resolution details with contextual information.

The U-Net's adaptability to satellite imagery segmentation enables precise localization of land boundaries and minimizes information loss. Its architecture efficiently captures both local and global features, making it a powerful tool for deriving insights from remote sensing data.

3.5 Transfer Learning

Transfer learning involves leveraging pretrained models trained on extensive datasets and adapting them to specific tasks. In this project, a U-Net architecture is first pretrained on the comprehensive *imagenet* database [4]. The pretrained model's parameters, denoted as θ_{pre} , capture general features from diverse images.

Fine-tuning is then conducted using Greenlandic satellite data. The model's parameters θ are adjusted through backpropagation to minimize a task-specific loss function. By initializing θ with θ_{pre} , the model starts with a set of non-random parameters that already possess a certain level of feature understanding. This accelerates model convergence, especially when dealing with limited high-quality data. The process can be mathematically represented as:

$$\theta^* = \arg \min_{\theta} \mathcal{L}_{\text{task}}(\theta) + \lambda \mathcal{R}(\theta)$$

where $\mathcal{L}_{\text{task}}$ is the task-specific loss, \mathcal{R} is a regularization term, and λ controls the trade-off between fitting the task and preventing overfitting.

Future research avenues could explore customized pre-trained models and transfer learning strategies tailored to Greenland's unique landscape characteristics, potentially enhancing the model's adaptability and performance.

3.6 GPU Cluster Efficiency

Training the U-Net model involves optimizing parameters using backpropagation and gradient descent. Complexity is approximated by operations per iteration.

CPUs handle large datasets and complex architectures sequentially, with time complexity $\mathcal{O}(N \cdot C \cdot W \cdot H \cdot d)$.

In contrast, GPUs excel in parallel processing, reducing complexity to $\mathcal{O}(\frac{N \cdot C \cdot W \cdot H \cdot d}{p})$, leveraging 10752 CUDA cores and 48GB VRAM on Nvidia RTX A6000.

For image dimensions ($N = 1000$, $C = 4$, $W = 512$, $H = 512$, $d = 5$) and an 8-core CPU with 64GB RAM versus Nvidia RTX A6000 GPU, CPU complexity is $\mathcal{O}(671,088,640)$, GPU complexity is $\mathcal{O}(163,840)$. GPUs offer high memory bandwidth and specialized hardware.

In conclusion, the GPU cluster with Nvidia RTX A6000’s 10752 CUDA cores and 48GB VRAM accelerates U-Net training. Parallelism and specialized hardware streamline model development.

4 Experiments and Results

4.1 Mosaic Creation

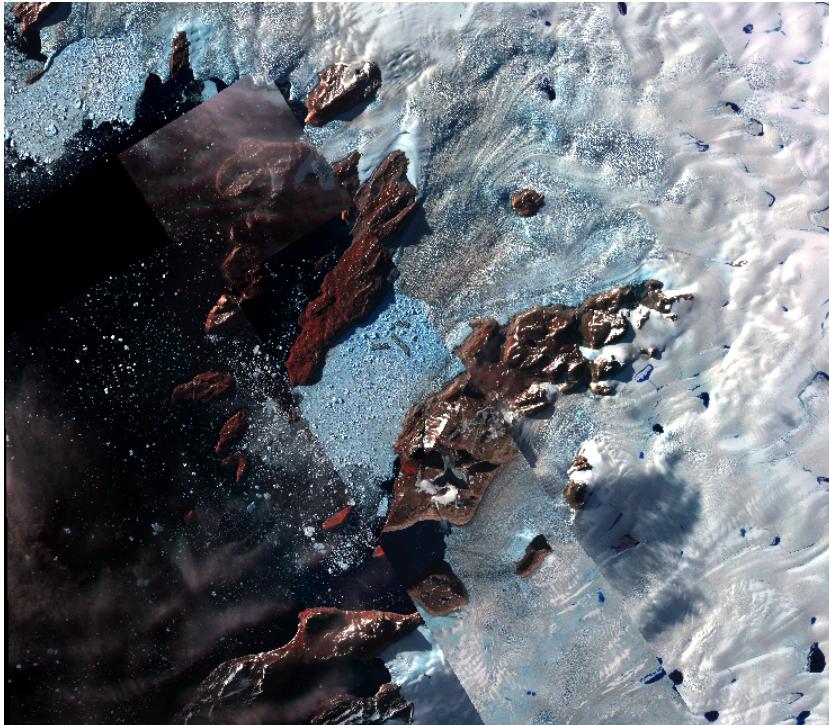


Figure 8: Mosaic created by the project using affine transformations and nearest neighbor interpolation. An artifact due to suboptimal alignment methods is visible, serving as a proof of concept. Image of the same area pre-processed by Planet is found in figure 1.

Planet’s pre-processing service efficiently generates mosaics by intelligently merging overlapping tiles, ensuring seamless and accurate results. This project initially attempted a custom mosaic creation method, but the outcomes were unsatisfactory compared to Planet’s service. The satellite images provided by Planet are discussed in 2.6.1.

Creating mosaics involves merging satellite tiles with precision, accounting for alignment, radiometric adjustments, and spatial coherence. A mosaic gen-

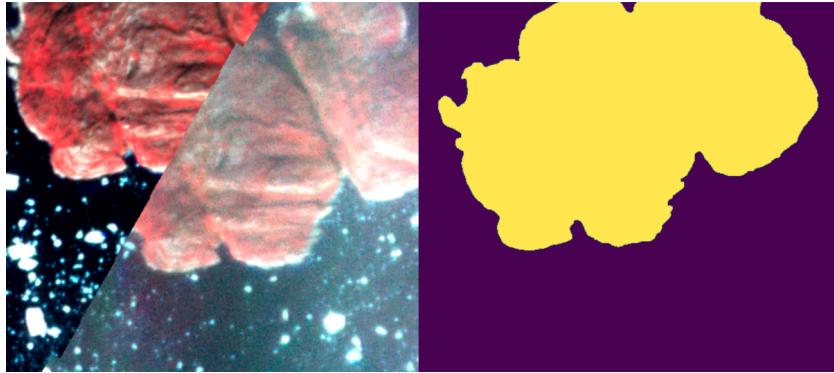


Figure 9: Small window of a satellite image with the corresponding mask. Notable alignment issues caused by translation errors are vividly highlighted.

erated using the author’s custom approach is depicted in Figure 8, revealing an artifact due to imperfect alignment methods. The resulting misalignment was considered too significant for proper registration.

4.2 Benchmarking

The U-Net model’s performance metrics, as detailed in subsection 3.2, are presented in Table 1. Figures 10, 11, 12, and 13 display sample input windows, true segmentation masks, and model predictions from the test set.

Metric	Value
Accuracy	.92
Precision	.72
Recall	.74
IoU	.53
Dice Score	.55

Table 1: Predictive Performance on Test Set

5 Discussions and Conclusions

5.1 Overall Performance

The evaluation metrics provide insights into the model’s performance in segmentation tasks. While the IoU, Dice, Precision, and Recall scores convey a consistent picture, they collectively indicate areas for potential enhancement.

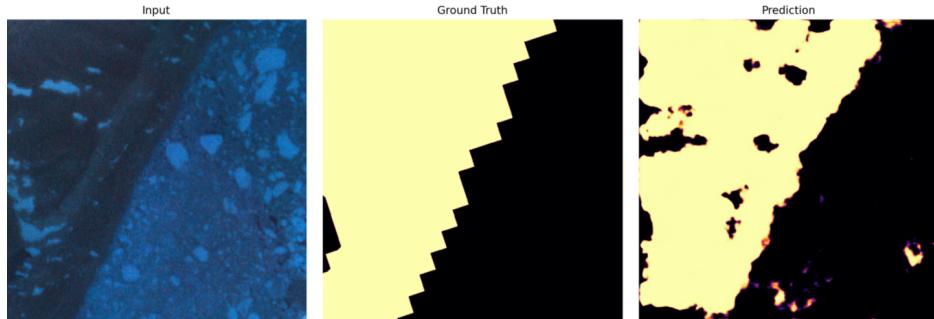


Figure 10: Input window (left), manually annotated ground truth (center), and prediction (right) from the test set.

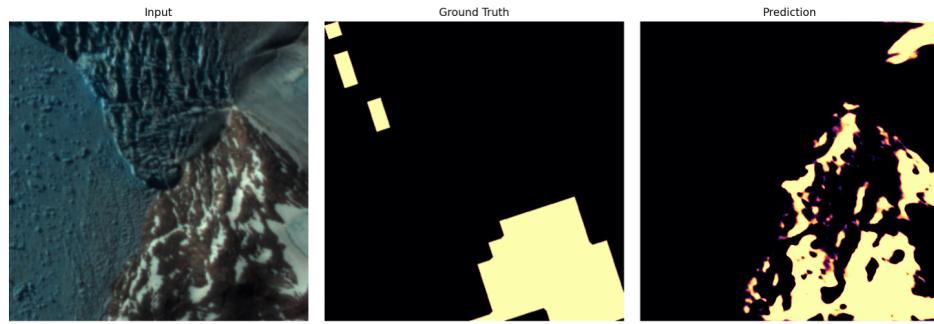


Figure 11: Input window (left), manually annotated ground truth (center), and prediction (right) from the test set.

The model demonstrates proficiency in delineating land boundaries, but challenges impacting mask quality and overall performance need to be addressed, especially in light of the limitations related to method comparisons.

5.2 Mask Quality and Limitations

5.2.1 Resolution and Human Error

Inherent bias introduced by variations in the target mask's resolution and inaccuracies from unknown satellite-image correspondence can lead to distortions in predictions and lower performance scores.

5.2.2 Expected Outcome and Proposed Solution

Given the limitations in the target mask's creation, a decrease in model performance was anticipated. To mitigate this, contributions to polygon creation details could improve mask evaluation and replication. Integrating

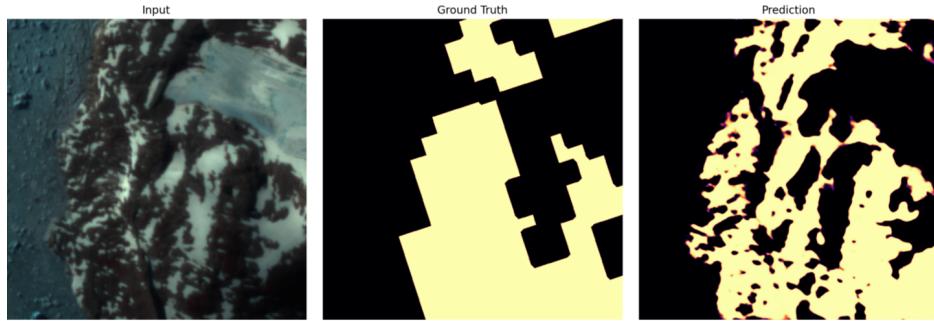


Figure 12: Input window (left), manually annotated ground truth (center), and prediction (right) from the test set.

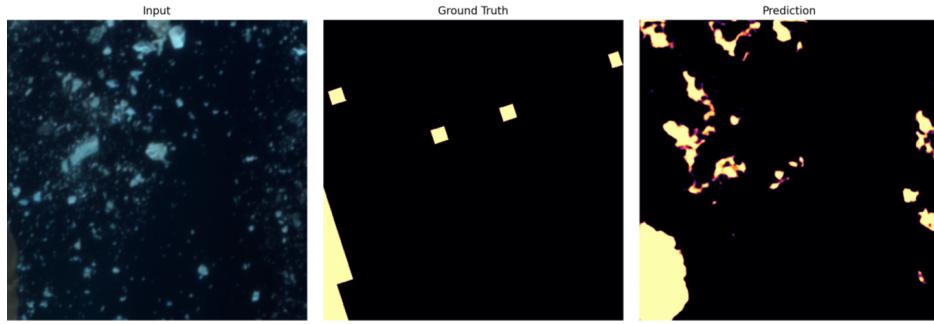


Figure 13: Input window (left), manually annotated ground truth (center), and prediction (right) from the test set.

high-resolution shapefiles derived from premium satellite imagery holds the potential to significantly enhance mask quality.

5.3 Limitations

5.3.1 Mosaic Creation

Manual tile merging for mosaic creation introduces misalignment, adversely affecting model performance, as exemplified in Figure 8. In contrast, Planet’s pre-processing service offers robust mosaic formation for more accurate analyses.

5.3.2 Data and Model Training

The model’s generalization is constrained by limited training areas and small datasets, further compounded by challenges arising from the mosaic creation

method. To mitigate these constraints, future endeavors should focus on expanding training areas and refining mosaic creation techniques.

5.4 Proposed Methods for Future Adaptation

5.4.1 Active Learning

Active learning selects informative samples iteratively for labeling to improve annotation efficiency. For satellite image segmentation, active learning can be expressed as:

$$\mathcal{D}_{\text{label}}^* = \arg \max_{\mathcal{D}_{\text{unlabel}}} \text{Uncertainty}(f, \mathcal{D}_{\text{unlabel}})$$

where $\mathcal{D}_{\text{label}}^*$ are chosen labeled samples, $\mathcal{D}_{\text{unlabel}}$ are unlabeled samples, and $\text{Uncertainty}(f, \mathcal{D}_{\text{unlabel}})$ quantifies model uncertainty.

5.4.2 Online Learning

Online learning adapts the model with incoming data. In this setup, model parameters θ_t are updated with each data point (x_t, y_t) :

$$\theta_{t+1} = \theta_t - \eta \nabla \mathcal{L}(f_{\theta_t}(x_t), y_t)$$

where \mathcal{L} is the loss function, η is the learning rate, and $f_{\theta_t}(x_t)$ is the model's prediction at time step t for input x_t .

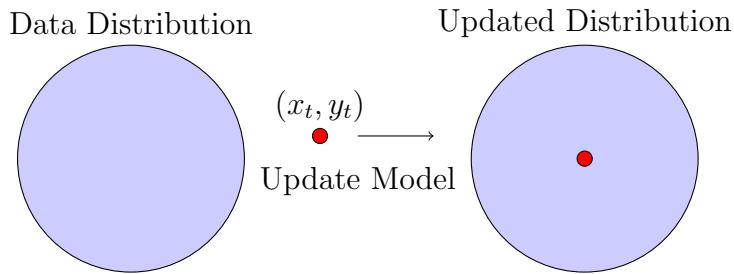


Figure 14: Illustration of Online Learning

Exploring active and online learning techniques within Greenland's land segmentation context could enhance model efficiency and adaptability for more accurate and up-to-date results.

Considering the unique dataset characteristics and the absence of direct method comparisons, investigating recent advancements like the Segment Anything Model (SAM) [7] could be a promising avenue. Inspired by the

large scale project known for the paper “An unexpectedly large count of trees in the West African Sahara and Sahel” by Brandt et al. [1], future work could implement active and online learning to improve segmentation performance and adaptability, addressing mask quality and alignment concerns.

Acknowledgements

The author extends heartfelt gratitude to François Bernard Lauze[5] for his invaluable guidance and support throughout the project. Special appreciation goes to Anders Anker Bjørk[3] and Jonas Kvist Andersen from the geography section for initiating the project idea and providing essential assistance in data gathering and interpretation.

The author would also like to thank Ankit Kariryaa, Stefan Oehmcke, and Christian Igel for generously sharing insights and information regarding the methods employed in their work ”An unexpectedly large count of trees in the West African Sahara and Sahel” [1].

Conflict of Interest

The author of this article declares no conflicts of interest, and affirms that the article has been authored solely by the author.

References

- [1] Martin Brandt et al. “An unexpectedly large count of trees in the West African Sahara and Sahel”. In: *Nature* 587.7832 (2020), pp. 78–82.
- [2] Johannes Brøns Christensen. *Automatic-Satellite-Island-Discovery*. Aug. 2023. URL: <https://github.com/JohannesBroens/Automatic-Satellite-Island-Discovery>.
- [3] University of Copenhagen. *Anders Anker Bjørk, Tenure Track Adjunkt, Staff*. 2023. URL: <https://orcid.org/0000-0002-4919-792X>.
- [4] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [5] University of Copenhagen Department of Computer Science. *Francois Bernard Lauze, Associate Professor, Staff*. 2023. URL: <https://di.ku.dk/english/staff/?pure=en/persons/200294>.
- [6] Environmental Systems Research Institute. “Shapefile technical description”. In: *An ESRI white paper* 4.1 (1998). URL: <https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf>.
- [7] Alexander Kirillov et al. *Segment Anything*. 2023. arXiv: 2304.02643 [cs.CV].
- [8] “Mass balance of the Greenland Ice Sheet from 1992 to 2018”. In: *Nature* 579.7798 (2020), pp. 233–239.
- [9] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems* 32. Curran Associates, Inc., 2019, pp. 8024–8035. URL: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
- [10] Planet Labs PBC. *GeoJSON and Areas of Interest (AOIs)*. © 2023 Planet Labs PBC. Planet, 2023. URL: <https://developers.planet.com/docs/planetschool/geojson-and-areas-of-interest-aois/>.
- [11] Planet Labs PBC. *Items and Assets*. © 2023 Planet Labs PBC. Planet, 2023. URL: <https://developers.planet.com/docs/apis/data/items-assets/>.
- [12] Planet Labs PBC. *Planet Application Program Interface: In Space for Life on Earth*. © 2023 Planet Labs PBC. Planet, 2023. URL: <https://api.planet.com>.

- [13] Planet Labs PBC. *Planet Explorer*. © 2023 Planet Labs PBC. Planet, 2023. URL: <https://www.planet.com/explorer/>.
- [14] Planet Labs PBC. *PlanetScope*. © 2023 Planet Labs PBC. Planet, 2023. URL: <https://developers.planet.com/docs/data/planetscope/>.
- [15] Planet Labs PBC. *Usable Data in Planet imagery*. © 2023 Planet Labs PBC. Planet, 2015. URL: <https://developers.planet.com/docs/planetschool/usable-data-in-planet-imagery/>.
- [16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.
- [17] H Seeger. “Spatial referencing and coordinate systems”. In: *Geographical information systems* 1 (1999), pp. 757–766.
- [18] Emmanuel Stefanakis. “Web mercator and raster tile maps: two cornerstones of online map service providers”. In: *Geomatica* 71.2 (2017), pp. 100–109.
- [19] Carole H Sudre et al. “Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations”. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*. Springer. 2017, pp. 240–248.
- [20] Byron D Tapley et al. “The gravity recovery and climate experiment: Mission overview and early results”. In: *Geophysical research letters* 31.9 (2004).
- [21] Saurabh Vijay et al. “Resolving seasonal ice velocity of 45 Greenlandic glaciers with very high temporal details”. In: *Geophysical Research Letters* 46.3 (2019), pp. 1485–1495.
- [22] Bao Zhang et al. “Geodetic and model data reveal different spatio-temporal patterns of transient mass changes over Greenland from 2007 to 2017”. In: *Earth and Planetary Science Letters* 515 (2019), pp. 154–163.

A Appendix

A.1 Planet Order

firstOrder

The screenshot shows a user interface for managing a Planet order. At the top, there's a navigation bar with a search bar containing "firstOrder". Below the search bar, there are two tabs: "OVERVIEW" and "Products". The "OVERVIEW" tab is selected, indicated by a blue border around its title. The "Products" tab has a grey border.

OVERVIEW

Name: firstOrder
Id: 65e63a6c-0e50-4daf-8337-f593697285ee
Created on: 2023-06-26 12:38:16 UTC

PRODUCTS

Running

Item type: PSScene

Asset types:

- ortho_analytic_4b_sr
- ortho_analytic_4b_xml
- ortho_udm2

Ordered via the "analytic_sr_udm2" product bundle

TOOLS

Clip

A preview image of a satellite scene showing a coastal area with a polygonal selection highlighted in red. Below the preview is a JSON object representing the polygon coordinates.

```
{"coordinates": [[[[-59.026999, 75.361278], [-57.105477, 75.356804], [-57.021744, 75.79759], [-59.107705, 75.785607], [-59.026999, 75.361278]]], "type": "Polygon"}
```

Products

Items

- 20220729_231346_30_220b
- 20220729_231348_37_220b
- 20220729_235020_33_227a
- 20220729_235022_61_227a
- 20220729_235024_89_227a
- 20220729_235027_16_227a
- 20220729_235029_44_227a
- 20220729_235232_78_24a5
- 20220729_235235_08_24a5
- 20220729_235237_37_24a5
- 20220731_154154_88_2499

Figure 15: Screenshot of the order from <https://www.planet.com/explorer/> © 2023 Planet Labs PBC.

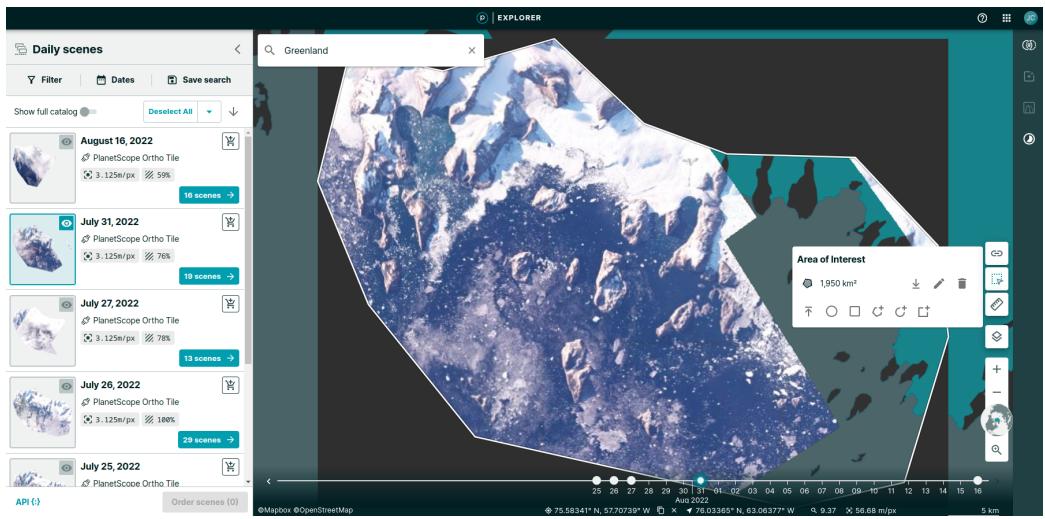


Figure 16: Screenshot of Planet Explorer <https://www.planet.com/explorer/> © 2023 Planet Labs PBC.

A.2 Different mosaic creation with mask

Mosaic with mask



Figure 17: Mosaic created by the project using affine transformations and the alternative linear interpolation. However, some tiles were excluded due to significant alignment issues. Manual mask alignment was also performed. The island mask is represented in white, which might not be the most suitable color choice.

A.3 Island Mask Zoomed

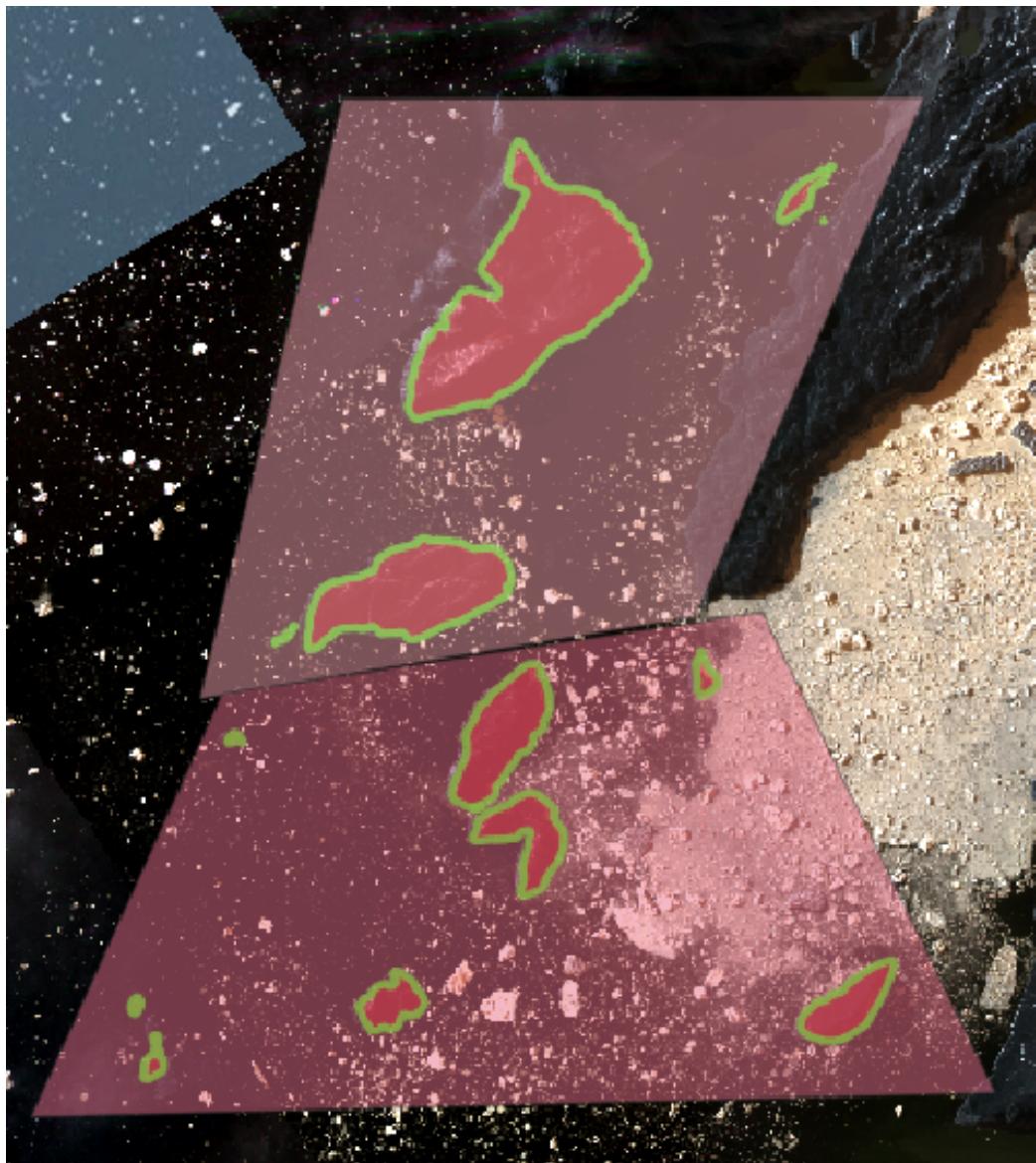


Figure 18: A screenshot taken in the beginning of the project, illustrating the masking of islands on a single tile in QGIS.

A.4 Project Description of the Contract

“This collaborative project between the Departments of Computer Science and Geography at the University of Copenhagen

aims to develop and compare image registration methods for high-resolution satellite images of Greenland. The primary focus is on registering previously unknown land areas under the ice sheets to understand their impact on rising water levels due to the climate crisis. The project will deliver a methods capable of dealing with satellite images taken at different times and resolutions. The final product will be a report and a presentation.”

Explanation of Deviations from Contract

Undertaken during holidays, this project encountered the challenge of limited real-time expert consultation. The code generously shared by Brandt et al. [1] was valuable, despite its lack of documentation, prompting the development of a new codebase. Due to time constraints, prioritization was essential, leading to the accomplishment of certain project goals. With gratitude for their contribution, this endeavor serves as a foundation for advancing land segmentation and climate impact analysis. Additionally, the final outcomes of this project include the creation of this report and a corresponding GitHub repository [2].