The Northern Change Lab

Mapping Supra-Glacier Rivers with Convolutional Neural Networks

Brown University

Authors:
Mason Lee

Abstract

This paper presents a novel application of dual U-Net architecture for high-resolution mapping of supra-glacial rivers across Greenland at an unprecedented 1m spatial resolution. The flexible dual U-Net approach addresses the long-standing challenge of river discontinuity often encountered in climate modeling, thereby overcoming the inherent limitations of traditional morphological operators. The first U-Net, RiverNet, translates satellite imagery into river segmentation maps, while the second, SegConnector, refines these maps by bridging discontinuous river segments. This iterative process ensures the generation of highly accurate and continuous river maps. We employ weakly supervised learning in data-sparse environments, leveraging transfer learning to improve performance. Custom loss functions, including Dice loss, masked Dice loss, and continuity loss, are introduced to handle weak labels and enforce topological continuity in the segmentation maps. The resulting high-resolution supra-glacial river maps are expected to provide significant insights into the melting dynamics of the Greenland ice sheet. Future work will focus on leveraging these maps to enhance our understanding of sea-level rise and ice sheet mass balance predictions. In addition, the enhanced river maps are anticipated to contribute to improvements in regional climate models, thereby providing more accurate and detailed data for environmental and climate change studies.

1 Introduction

The future trajectory of global climate change and subsequent sea-level rise is a topic of great concern and remains a highly active area of research. One crucial component of this equation is the meltwater runoff from the Greenland Ice Sheet (GrIS). Current estimates suggest that GrIS runoff contributes significantly to global sea level rise, yet substantial uncertainty persists in its accurate measurement and prediction [**?**].

A common approach for estimating ice sheet runoff is in situ gauging of proglacial rivers that drain the ice sheet, coupled with surface mass balance (SMB) modeling. However, these methods have their limitations, especially in terms of scale and accuracy. Recent work on the Inglefield Ice Sheet highlighted these shortcomings, showing discrepancies in runoff measurements [**?**].

One promising avenue to address these challenges is through the use of remote sensing. The advent of high-resolution satellite imagery such as WorldView, Landsat, and Sentinel-1, with spatial resolutions of 1m, 30m, and 10m respectively, has revolutionized our ability to image the Arctic. These platforms provide an unprecedented level of detail, opening up new possibilities for studying glacial and proglacial environments.

Yet, despite these advancements, mapping supra-glacial rivers remains a considerable challenge. The complex geometries, transient nature of rivers, presence of moulins, and lack of high-resolution imagery in some regions contribute to the difficulty. Furthermore, the fractal geometry of river networks means that accurate delineation requires not just high resolution, but also a sophisticated understanding of river morphology.

Against this backdrop, we propose the use of Convolutional Neural Networks (CNNs) to map supra-glacial rivers. CNNs have emerged as a powerful tool in computer vision and have seen rapid advancements in recent years, driven by the open-source community. In remote sensing, these deep learning techniques offer the potential to extract complex features from high-resolution satellite imagery and accurately map river networks.

In this study, we leverage these advancements to create the most in-depth map of a supra-glacial river network to date. The implications of this work extend beyond glaciology, providing a foundation for further advancements in remote sensing and climate modeling.

1.1 Regional Climate Models in Greenland

The accurate prediction of Greenland's ice sheet dynamics is a complex task that relies heavily on the use of climate models. These models, which are physics-based, make a series of limiting assumptions to simplify the real-world processes involved in glacial dynamics. However, these simplifications can sometimes lead to inaccuracies in prediction and forecasting.

One such model widely used is the Modèle Atmosphérique Régional (MAR), a regional atmospheric climate model specifically developed for Greenland's unique climate [**?**]. MAR has been instrumental in studying Greenland's climate dynamics and the associated meltwater runoff. However, it is not without its limitations. One persistent issue with the MAR model, as with many climate models, is the problem of lagged delay. That is, the model's predictions do not always align temporally with the observations.

This discrepancy is particularly noticeable in the case of runoff measurements. There are several runoff gauges across Greenland that collect real-time data on meltwater runoff. Comparisons between these measurements and the predictions made by the MAR model often show an overestimation of runoff by the model [**?**].

Some scientists propose that this discrepancy may be due to the transportation of water off the ice sheet through the supra-glacial river network. This hypothesis, however, is not thoroughly understood, primarily due to the difficulties in accurately mapping and monitoring these river networks. This study aims to address this knowledge gap by developing a more accurate method of mapping supra-glacial rivers using CNNs, potentially contributing to the improvement of regional climate models like MAR.

1.2 Harnessing the Power of Convolutional Neural Networks: U-Net and Transfer Learning

Convolutional Neural Networks (CNNs) have revolutionized the field of computer vision due to their capacity to automatically learn intricate and abstract feature representations from image data. The defining characteristic of CNNs is their ability to exploit spatial correlations within an image through the application of convolutional filters. These filters are learned from the data during the training phase and are capable of capturing local dependencies and hierarchical patterns within the image, making CNNs particularly potent tools for image classification and segmentation tasks [**?**].

A notable development in the realm of CNNs is the U-Net architecture, which was initially conceived for the purpose of biomedical image segmentation, where the availability of labeled training data is typically limited [**?**]. The architecture of U-Net is composed of two primary segments: the encoder (or contracting) path and the decoder (or expansive) path. The encoder employs successive convolutional and max pooling layers to reduce spatial dimensions while increasing depth, extracting and encapsulating high-level features from the input image. The decoder, conversely, consists of upconvolutional (or transposed convolutional) layers followed by concatenation and standard convolutions, enabling it to recover spatial information and provide precise localization. The skip connections between corresponding layers of the encoder and decoder are a distinctive feature of U-Net, allowing it to retain and utilize high-resolution features for reconstruction, thus enhancing the precision of the output.

The concept of transfer learning has been instrumental in the successful application of deep learning models, particularly in scenarios where labeled data is limited. The core idea behind transfer learning is the utilization of a pre-trained model, typically trained on a large and diverse dataset, and fine-tuning it for a specific task. This practice leverages the rich feature representations learned from a wide variety of image data, and it's particularly beneficial when the task-specific dataset is not sufficiently large to train a deep model from scratch.

In the context of this study, we employ a U-Net model pretrained on the ImageNet dataset [**?**], a large-scale dataset encompassing over a thousand diverse categories. The premise behind the transferability of a model pretrained on ImageNet to the task of mapping supra-glacial rivers lies in the universal feature-learning ability of CNNs. The lower layers of a pretrained CNN typically learn low-level features, such as edges and textures, which are common across various tasks. These learned features are transferable and can be utilized as a meaningful initialization for the task-specific model, which is then fine-tuned to learn the higher-level features pertinent to the target task. Given the complex textures and morphological variations within glacial environments, we posit that this transfer learning approach will provide a robust starting point for our model and will ultimately lead to an accurate and precise mapping of supra-glacial rivers.

1.3 Challenges in Supra-Glacial River Segmentation and the Promise of CNNs

Segmenting supra-glacial rivers from high-resolution satellite imagery is far from a trivial task. Several factors contribute to the complexity of this problem. Firstly, there is a dearth of segmented datasets at 1m resolution, with the exception of a small dataset curated by our lab. This scarcity of labeled data makes it challenging to train supervised learning models, which typically require substantial amounts of annotated data.

Secondly, a lack of expert domain knowledge in segmentation can lead to inaccuracies in the manual labeling process, such as over-segmenting or under-segmenting rivers, or missing moulins. These inaccuracies can subsequently bias the learning of the model and impact its performance.

Thirdly, the characteristics of the ice sheet can vary dramatically across different regions, requiring a model that is not only accurate but also robust and adaptable to these changing conditions.

Lastly, supra-glacial river networks are highly discontinuous. They often start and stop abruptly, with many terminating beneath the ice sheet. This discontinuity adds another layer of complexity to the task of accurately mapping these river networks.

Despite these challenges, recent research suggests that CNNs, particularly those utilizing the U-Net architecture, are capable of tackling such complex segmentation tasks. A growing body of work in the field of remote sensing has demonstrated the effectiveness of CNNs in extracting meaningful features from satellite imagery across a variety of tasks and conditions [**?**].

The rapid pace of advancements in the field of computer vision, fueled by the open-source community, has resulted in increasingly sophisticated and efficient CNN architectures. These advancements, combined with the inherent ability of CNNs to generalize from learned features, make them a promising tool for our task.

The flexibility and adaptability of deep learning models, such as CNNs, are also key strengths. They are capable of learning to identify and extract the most relevant features for a given task, making them ideally suited for environments where the input data can vary widely, as is the case with our ice sheet imagery.

Lastly, the ability to leverage transfer learning further amplifies the potential of CNNs. By initializing our model with weights learned from a large and diverse dataset like ImageNet, we can overcome the limitations imposed by our relatively small dataset and kick-start the learning process of our model, increasing its chances of success in accurately mapping supra-glacial rivers.

2 Methods

2.0.1 Data Preprocessing

The data preprocessing pipeline is a vital part of our methodology. It transforms raw satellite imagery into suitable input for our U-Net model and ensures that the model's output can be interpreted correctly.

Image Collection and Tiling We use WorldView satellite images with a spatial resolution of 1 meter. To fit these high-resolution images into our model, we divide them into smaller tiles of size 512x512 pixels. This tiling operation is performed in a way that ensures the complete coverage of the original image without any overlap between the tiles.

Encoding Although WorldView images are in grayscale, our U-Net model requires input with three channels. Therefore, we convert the grayscale images to RGB by copying the grayscale channel into the other two channels. This results in an RGB image where all three channels are identical.

Data Augmentation To increase the size and diversity of our dataset, we perform several data augmentation operations. These include flipping the images horizontally and vertically, adding random noise, blurring, adjusting the contrast, and altering the hue and saturation. These augmentations help our model generalize better to unseen data.

Auxiliary Bands In addition to the RGB channels, we incorporate two auxiliary bands into our model. These bands are derived from the Normalized Difference Water Index (NDWI) calculation. They provide additional context for the model, helping it distinguish between water and non-water elements in the images.

Post-processing Once the model generates its predictions, we apply a series of post-processing operations to refine the output. These include path opening and closing operations to remove small gaps and bridges in the predicted rivers, noise filtering to remove small, isolated predictions, and thresholding to convert the model's continuous output into a binary segmentation map. The post-processing stage ensures the model's output is clean and suitable for further analysis in our study.

2.1 U-Net Architecture

The U-Net architecture is a convolutional neural network (CNN) specifically designed for image segmentation tasks. It consists of two main parts: the encoder and the decoder, which work together to capture and then utilize information at different scales in the image.

2.1.1 Encoder

The encoder follows the traditional architecture of a CNN. It is composed of multiple convolutional layers that are designed to extract and learn a hierarchy of features from the input image. Each layer in the encoder comprises two parts: a convolution followed by a rectified linear unit (ReLU) for introducing non-linearity, and a max pooling operation for spatial down-sampling. Let f_i represent the feature map at the i-th layer of the encoder. The operation at each layer of the encoder can be represented by the equation:

$$f_i = \mathrm{ReLU}(\mathrm{Conv}(f_{i-1})),$$

where Conv is the convolution operation.

2.1.2 Decoder

The decoder essentially mirrors the operations of the encoder but in reverse order, hence the U-shape of the U-Net architecture. The aim is to recover the spatial information that is lost during the encoding process. Instead of max pooling operations, the decoder uses transposed convolutions, also known as deconvolutions, for spatial up-sampling. The operation at each layer of the decoder can be represented by the equation:

$$f_i = \mathrm{ReLU}(\mathrm{Deconv}(f_{i-1})),$$

where Deconv is the deconvolution operation.

2.1.3 Skip Connections

A unique feature of U-Net is the inclusion of skip connections, also known as residual connections, which pass the output from each layer of the encoder directly to the corresponding layer in the decoder. This helps to retain the high-resolution features from the input image that might be lost during the encoding and decoding process. Formally, the operation at each layer of the decoder becomes:

$$f_i = \mathrm{ReLU}(\mathrm{Deconv}(f_{i-1} + f_{i_{\mathrm{encoder}}})),$$

where $f_{i_{\mathrm{encoder}}}$ is the output from the corresponding layer in the encoder.

2.1.4 Flexibility

The U-Net architecture is inherently flexible. The depth of the network (i.e., the number of layers in the encoder/decoder) can be adjusted depending on the size and complexity of the input images. Moreover, the number of filters in each convolutional layer can be increased or decreased to control the capacity of the network. This makes U-Net adaptable to a wide range of image segmentation tasks.

2.2 Training the U-Net Model

The U-Net model was trained using a Adam Optimizer, with a learning rate of 0.001. The learning rate was reduced by a factor of 10 after every 50 epochs. The model was pretrained on the ImageNet dataset and then fine-tuned on our task-specific dataset using the varying loss functions described below. We used a batch size of 16 and trained the model for 100 epochs.

To account for the small size of our dataset and prevent overfitting, we utilized extensive data augmentation, including rotations, scaling, and elastic deformations.

2.3 Loss Functions

2.3.1 Dice Loss

The Dice coefficient, also known as the Sørensen–Dice coefficient, measures the similarity between two samples. For binary classification, it can be defined as:

$$L_{\text{Dice}} = 1 - \frac{2 \times \text{Number of overlapping pixels} + \text{Smooth}}{\text{Total number of pixels in both maps} + \text{Smooth}} \tag{1}$$

where $\mathbf{p}$ and $\mathbf{g}$ are the predicted and ground truth binary segmentation maps respectively, N is the total number of pixels, and ϵ is a small constant to avoid division by zero. The Dice loss is then defined as $1 - D(\mathbf{p}, \mathbf{g})$.

2.3.2 Masked Dice Loss

When dealing with weakly-supervised data, the Dice loss can be modified to only consider the masked regions of the input. The mask is denoted by $\mathbf{m}$ and is applied element-wise to both $\mathbf{p}$ and $\mathbf{g}$ before calculating the Dice coefficient. The masked Dice loss is then defined as:

$$L_{\text{Masked Dice}} = 1 - \frac{2 \times \text{Number of overlapping pixels in masked region} + \text{Smooth}}{\text{Total number of pixels in masked region} + \text{Smooth}} \tag{2}$$

2.3.3 Continuity Loss

The continuity of the predicted segmentation map is critical in the context of our problem as river networks are often discontinuous and exhibit sudden starts and stops. To ensure smoothness and continuity in the predicted segmentation, we incorporate a continuity loss that penalizes short lines and noise. The continuity loss is composed of a single residual term and several experiments were run with different continuity losses with different properties: total variation loss, Laplacian loss, and line length loss.

Total Variation Loss The total variation (TV) loss encourages spatial smoothness in the image. It does this by measuring the absolute differences between neighboring pixel values, effectively penalizing rapid changes in pixel intensities. Thus, the TV loss serves as a regularizer that smooths the output by minimizing these differences. For a 2D image, it is formally defined as:

$$L_{\text{TV}} = \text{Sum of absolute differences of neighboring pixels} \tag{3}$$

where N and M are the height and width of the image respectively, and $p_{i,j}$ denotes the pixel value at location (i, j).

Laplacian Loss The Laplacian loss penalizes high-frequency components in the image, effectively reducing noise and promoting smoother transitions between different regions. The Laplacian of an image is a measure of the second order derivatives, capturing the rates of change in pixel intensities. This property makes the Laplacian effective for edge detection and, in our case, for identifying discontinuities in the river networks. The Laplacian loss is defined as the sum of the absolute Laplacian values:

$$L_{\text{Lap}} = \text{Sum of absolute second derivatives of pixels} \tag{4}$$

where again N and M are the height and width of the image respectively, and $p_{i,j}$ denotes the pixel value at location (i, j).

Line Length Loss The line length loss is designed to penalize short contours in the predicted segmentation. The rationale behind this loss is the assumption that real river networks tend to form longer, continuous lines rather than a multitude of short, disconnected segments. To express this mathematically, we compute the inverse of the average line length in the binary prediction, squared. This encourages the model to produce longer, uninterrupted river predictions:

$$L_{\text{LL}}(\mathbf{p}) = \left(\frac{1}{\frac{1}{N_{\text{lines}}} \sum_{i=1}^{N_{\text{lines}}} l_i} \right)^2 \tag{5}$$

where N_{lines} denotes the number of distinct lines (contours) in the predicted segmentation, and l_i is the length of the i-th line.

2.3.4 Dual U-Net Architecture

In order to address the diverse challenges posed by the sparse data environment in Greenland and aid scientists with auxiliary tasks, we introduce a dual U-Net architecture. Despite sharing the same fundamental architecture, each U-Net serves a unique purpose and processes different input data.

RiverNet: Satellite-to-Segmentation U-Net The first U-Net, which we name RiverNet, is designed to translate satellite imagery directly into river segmentation maps. Given an input satellite image, RiverNet produces an output segmentation map that delineates the rivers in the image. The transformation can be represented by the following equation:

SegConnector: Discontinuity-to-Continuity U-Net The second U-Net, named SegConnector, takes in a discontinuous river segmentation map and refines it by connecting discontinuous lines. It is especially useful when the output from RiverNet or any other segmentation process contains fragmented or incomplete river paths. Given a discontinuous segmentation map, SegConnector generates a continuous segmentation map that connects the fragmented segments. One significant advantage of SegConnector is that it can be run iteratively at different resolutions to enforce various degrees of line connectivity. This flexibility makes it a more robust tool for achieving continuity in river segmentation maps compared to traditional path-opening methods

2.3.5 Full Prediction on a GeoTIFF

The dual U-Net pipeline enables us to perform comprehensive predictions on large GeoTIFF files, making it a robust tool for extracting river networks from satellite imagery. The prediction process involves several steps, each contributing to the refinement of the output segmentation map.

Ensemble RiverNet Prediction The first step in the process involves the RiverNet ensemble. We load multiple versions of RiverNet weights, each representing different stages of the training regime. This ensemble of k models , predicts the river segmentation map. The prediction from each model is thresholded by its training epoch, which intuitively means that models trained for fewer epochs predict more segmentation features, while models trained for more epochs are more selective. Therefore, regions that are predicted by multiple models have increased width and exhibit greater continuity. This can be represented mathematically as:

$$S_{\text{ensemble}} = \text{Threshold}\left(\frac{1}{k} \sum_{i=1}^{k} \text{RiverNet}_i(I) \right) \tag{6}$$

Noise Removal and Path Closing The next stage involves post-processing of the ensemble prediction. This includes noise removal, path closing, and thresholding operations to refine the segmentation map, eliminate spurious segments, and ensure the continuity of the river network.

Iterative SegConnector Refinement Subsequently, the SegConnector model is applied iteratively, 'n' times, to further enhance the continuity and connectivity of the river segments. A final post-processing step removes lines shorter than a certain threshold, ensuring that only significant river segments are retained.

Large GeoTIFF Handling The developed pipeline is capable of handling up to 150GB of concurrent GeoTIFF data. Leveraging the rasterio environment, it can tile and process large images with variable bands effectively. The U-Net architectures, built with TensorFlow, form the backbone of the prediction models, enabling efficient and accurate segmentation of river networks from high-resolution satellite imagery.

2.4 Semi-Supervised Learning

Semi-supervised learning is a machine learning paradigm that makes use of both labeled and unlabeled data during the training process. This approach is especially effective when the amount of available labeled data is limited. It allows us to leverage the unlabeled data to enhance the learning process and improve the model's performance.

In our pipeline, the semi-supervised learning process consists of several steps:

2.4.1 Initial Training on Labeled Data

The first step in our pipeline is to train the U-Net model on the available labeled data. This stage serves to establish an initial model that is capable of performing the task of river segmentation, albeit potentially imperfectly or incompletely.

2.4.2 Post-Processing Techniques

After the initial training, we keep part of the original training set as a validation set for the ongoing model evaluation. We then implement various post-processing techniques to refine the model's output. These techniques include running SegConnector to enforce continuity in the segmentation map, applying path-opening methods, and removing small objects and noise.

2.4.3 Weak Supervision

The model's predictions on the unlabeled data, after being processed through the above steps, serve as weakly supervised labels. These labels, though they may not be perfect, still contain valuable information that can be used to further improve the model.

2.4.4 Training with Masked Dice Loss

The model is then trained again, this time incorporating both the original labeled data and the weakly supervised labels. The masked Dice loss is applied during this stage, enforcing the loss only at the predicted labeled points and masking out everything else. This approach allows the model to learn from its previous predictions and to enhance its ability to generalize to new data.

2.4.5 Iterative Learning

This entire process—post-processing, weak supervision, and training with masked Dice loss—is repeated iteratively, with each iteration serving to further improve the model's performance. By incorporating the model's own predictions into the learning process, we effectively use the model to teach itself, enabling it to continually improve its ability to accurately segment rivers in satellite imagery.

2.5 Additional Methods and Failed Experiments

In our pursuit of an effective model for river network extraction, we experimented with various techniques and strategies, some of which yielded positive results while others did not meet our expectations. This section provides an overview of these additional methods and failed experiments.

Loop Contour Detection We developed an algorithm using OpenCV2 to detect and remove loop contours from the segmentation maps. These loop contours often signify errors in the extraction process, and their removal can help improve the quality of the final river network.

Path-Opening in Loss Function We experimented with integrating path-opening operations into the loss function. The idea was to remove certain objects from the segmentation maps and then penalize the difference between the modified and original maps. This method was designed to encourage the model to avoid producing these undesirable objects in the first place.

Despite these promising methods, several of our experiments did not yield the expected results:

Continuity Loss with Contour Size and BCE An attempt was made to enforce continuity by incorporating the absolute size of contours and binary cross-entropy into the loss function. However, this approach did not effectively improve the continuity of the extracted river networks.

GANs for Synthetic Data We explored the possibility of using Generative Adversarial Networks (GANs), specifically CycleGAN, to generate synthetic river networks. Unfortunately, the synthetic data did not contribute meaningfully to improving the performance of our models. This is mostly likely due to the need for large datasets in adversarial architectures.

Generative Inpainting Techniques Our exploration of generative inpainting techniques, such as diffusion models, to fill gaps in large TIFFs also fell short of expectations. Despite the promise of these techniques, they did not effectively enhance the connectivity of the river networks.

Variational Autoencoder for Inpainting Finally, we considered using a Variational Autoencoder (VAE) to inpaint river connections. The idea was to condition the VAE on different connection thresholds, enabling the generation of connections of varying degrees from the latent space. However, this approach did not produce the desired results, as the VAE failed to create reliable and realistic river connections.

3 Results

3.1 Results

Our proposed models, RiverNet and SegConnector, showed promising results, displaying significant improvements over existing methods in the extraction and connection of river networks from satellite images. Each figure discussed below demonstrates various aspects of the models' performance.

Figure 1 showcases an early challenge faced in the development of SegConnector: the generation of loops while connecting discontinuous segments. Loops are common in natural river networks, but they pose complications for river routing processes. Through iterative modifications to data pre and post-processing methods, we managed to eliminate the occurrence of such features, leading to cleaner and more useful river network predictions.

The improvements in SegConnector's predictions can be observed in Figure 2. The model was trained on images of varying resolutions, demonstrating its adaptability to different scales. This scalability is a valuable feature for processing high-resolution satellite data where river segments can span vastly different scales.

Figure 3 illustrates the application of Masked Dice Loss during the training of RiverNet. This loss function only enforces penalties at points within the mask, effectively allowing over-prediction and providing a more robust prediction. By creating a buffer of arbitrary size around positive values, we can enforce true negatives. This figure highlights how under-segmented input images can be effectively segmented using this weak supervision technique.

Figure 4 provides a visualization of a segmentation map generated by RiverNet and further refined by SegConnector, overlayed on the original training geoTIFF. The ground truth segmentation is represented in black, while the output of a single RiverNet pass is shown in yellow. The red lines represent the segmentation map after the application of SegConnector, which connects discontinuous lines, providing a more accurate river network extraction.

Lastly, Figure 5 demonstrates the power of our model ensemble. By incorporating feature representations from early epochs, the user can effectively control the sparsity of the rivers. This flexibility is key to achieving the desired level of detail in the extracted river networks, accommodating various use cases and user preferences.

These results underline the potential of RiverNet and SegConnector as powerful tools for river network extraction and connection, respectively. They offer a novel solution to the challenges faced in river

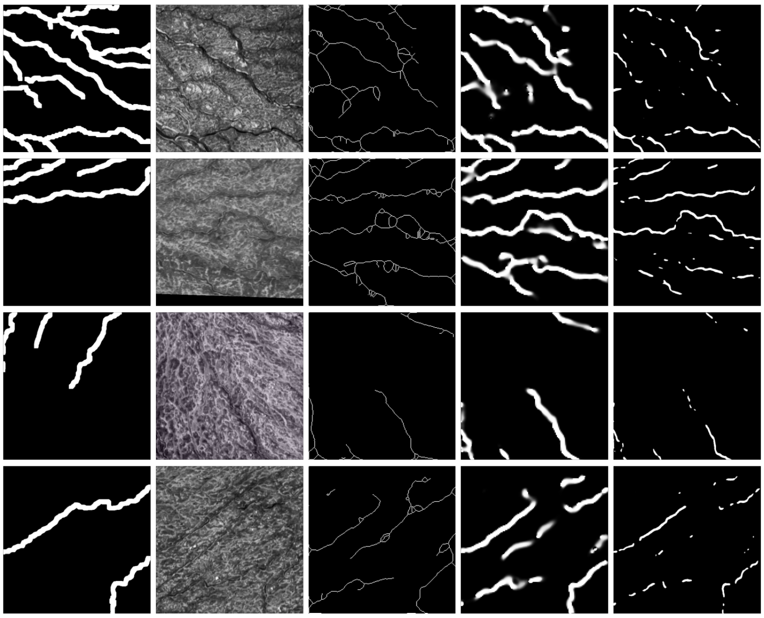

Figure 1: Early on, the use of SegConnector was made difficult by the prediction of loops in the connection of discontinuous segments. The presence of loops within river networks is common but for the purpose of river routing, should be avoided. Modifications to data pre and post processing has removed the existence of these feautures

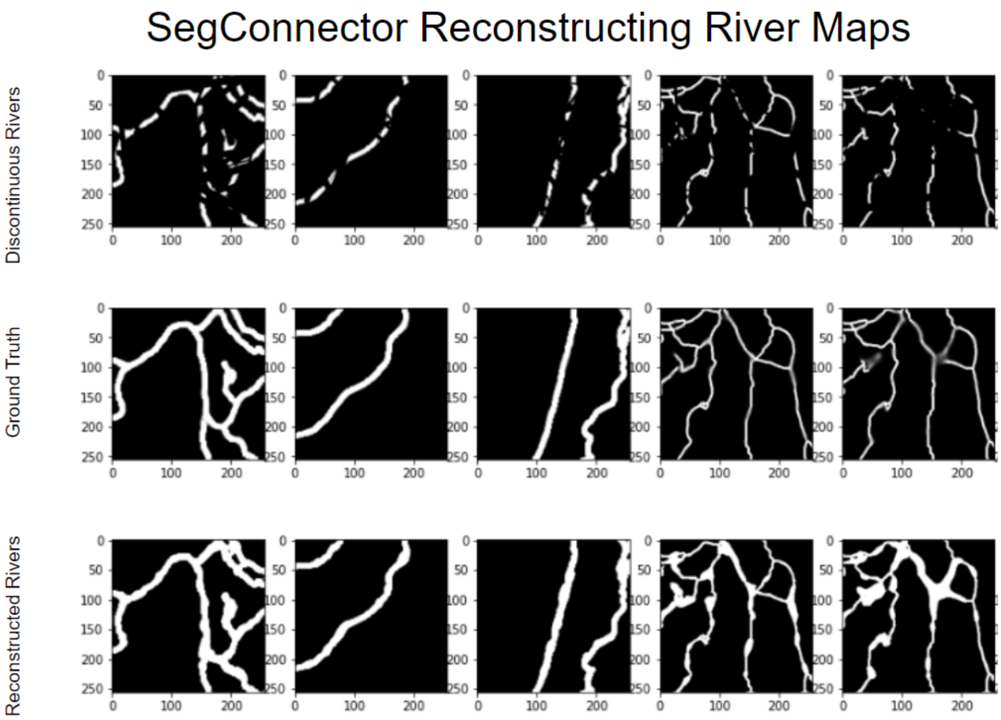

Figure 2: Improved Predictions of SegConnector. The model is trained on varying resolutions of images and can handle reconstruction at multiple scales

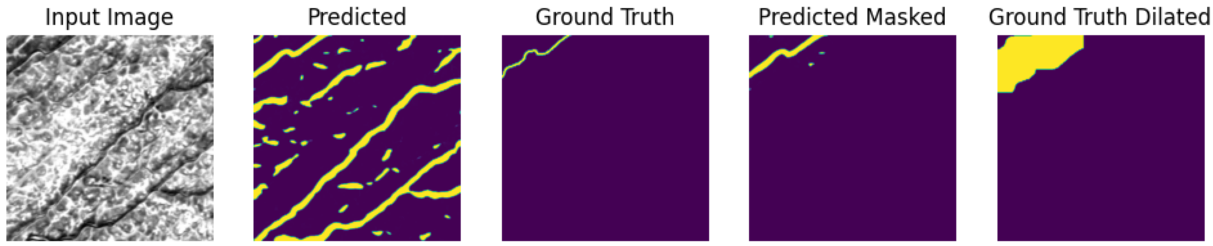

Figure 3: Masked Dice Loss during training of RiverNet. The loss is only enforced at points within the mask, over-prediction is not penalized which allows for a more robust prediction. A buffer of arbitrary size is created around the positive values, allowing the network to enforce true negatives. The under-segmented input image can be effectively segmented using this weak supervision technique.

network extraction from satellite imagery, and their performance suggests potential for further enhancements and application in other related domains.

3.2 Discussion

The techniques employed in this study have shown substantial success in addressing data sparsity, a common issue in remote sensing and specifically in the extraction of river networks from satellite images. By developing and utilizing two distinct models, RiverNet and SegConnector, we have been able to extract meaningful information from sparse data and generate connected, continuous river network predictions.

RiverNet, with its ensemble approach and the usage of Masked Dice Loss, showed robust performance in extracting river networks, even in the presence of sparse or incomplete data. SegConnector, on the other hand, successfully tackled the challenge of connecting these discontinuous river segments, generating more usable and accurate river network maps. Together, these models offer a comprehensive solution to a complex problem.

However, while the current results are encouraging, there is still considerable work to be done to improve the models' predictions. Tuning the models to be more user-specific based on their needs and preferences is a crucial area for future work. This includes refining the balance between over-segmentation and under-segmentation, as well as the degree of connectivity in the final river network maps. The algorithm still underpreforms in certain regions of Greenland, under-segmenting images.

Furthermore, it is essential to ensure that our work is accessible and usable by the wider scientific community. We plan to publish the code and provide comprehensive documentation to enable other researchers to build upon our work. This also involves a significant refactoring effort to improve the efficiency, readability, and modularity of the codebase.

Our findings underline the potential of deep learning techniques in processing sparse data and extracting meaningful information from it. The successful application of these techniques in this study opens up possibilities for further research in similar domains.

3.3 Failures

While our proposed methodology has achieved significant milestones in river network extraction from satellite imagery, we acknowledge several limitations and areas for improvement.

Firstly, identifying the optimal parameters for our models has proved challenging. Depending on the specific region in Greenland, the output images are often either over-segmented or under-segmented. This inconsistency in performance across different geographical areas suggests the need for a more adaptive or region-specific parameter tuning strategy. Additionally, the presence of grid patterns in some output maps points towards an area of improvement in our pre-processing or model architecture.

Secondly, the current implementation of the SegConnector model is not as robust to noise as desired. This shortcoming is primarily attributed to certain missteps during the model's training phase. Given the important role SegConnector plays in ensuring continuity in the extracted river networks, it is critical to address this limitation in future iterations. Refining the training process and potentially incorporating noise reduction techniques could enhance SegConnector's performance and resilience to noise.

Thirdly, RiverNet has demonstrated difficulty in handling rocky areas within the satellite imagery. These regions present unique challenges due to their complex textures and contrast levels, which the

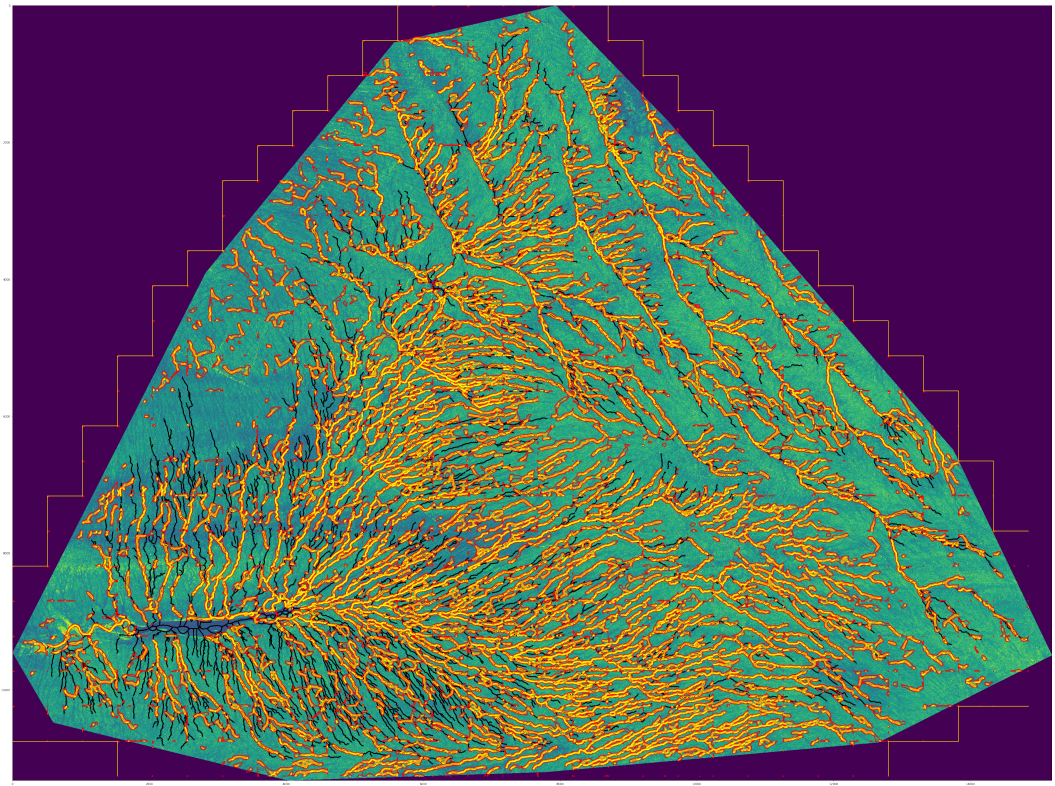

Figure 4: RiverNet + SegConnector segmentation map overlayed on the original training geoTIFF. Black represents the ground truth segmentation. Yellow represents one pass of RiverNet segmentation. Red represents the segmentation map after connecting discontinuous lines.

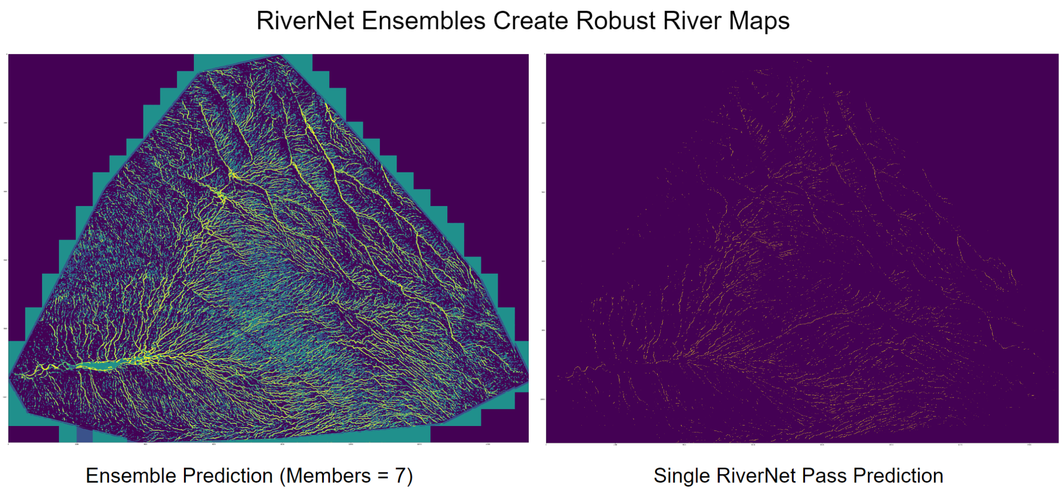

Figure 5: Segmentation maps with Model Ensemble. Incorporating feature representations from early epochs allows for the user to control the sparsity of the rivers.

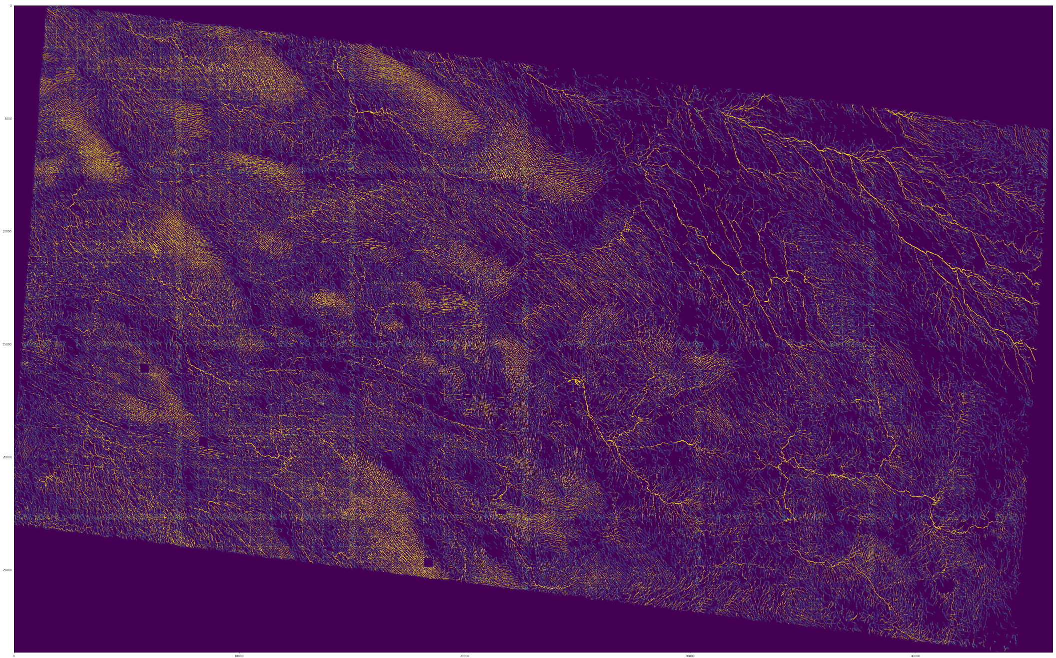

Figure 6: RiverNet + SegConnector segmentation map overlayed on the original training geoTIFF. Black represents the ground truth segmentation. Yellow represents one pass of RiverNet segmentation. Red represents the segmentation map after connecting discontinuous lines.

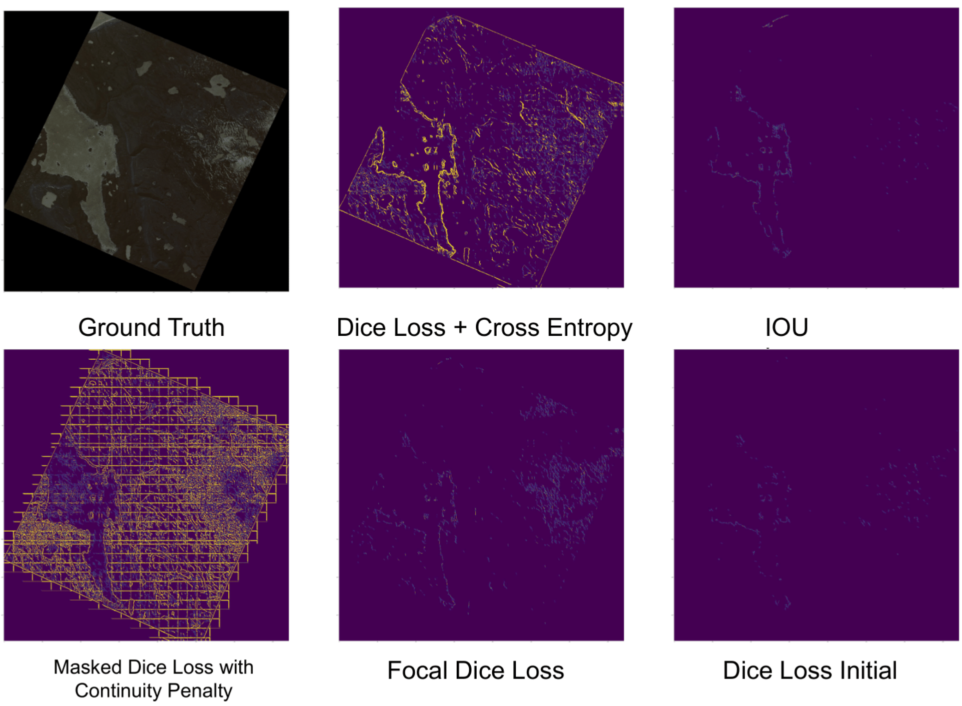

Figure 7: The effect of different loss function on the river map segmentations on a rocky region on the Minturn Eastern-Boundary. Without examples of this region, RiverNet struggles to predict accurately.

model currently struggles to accurately interpret. Moreover, the overall pipeline exhibits sensitivity to different normalization and standardization techniques, highlighting a need for further robustness.

Despite these limitations, it is worth noting the significant progress that has been made in such a data-sparse environment. The acquisition of the new data set from the Buckley School will be a critical step in further improving and refining our algorithm.

As shown in 7, our iterative approach to model development and refinement has consistently led to improvements in our output. Each iteration provides valuable insights and lessons, guiding our next steps and informing future improvements.

In conclusion, while acknowledging our current limitations, we remain optimistic about the potential of our methodology. With further refinement and the integration of new data, we are confident in our ability to continue enhancing the accuracy and utility of our river network extraction models.

3.4 Future Work

The dual U-Net pipeline offers a promising approach to river network extraction from satellite imagery. However, we acknowledge that there are several potential improvements and expansions that could further enhance its performance and applicability.

- Expanded Dataset: Our forthcoming collaboration with the Buckley School will significantly increase our dataset, offering a wider range of labeled examples for training and testing our models. This will not only improve the robustness of our models but also potentially enhance their generalization capabilities.
- Exploring Advanced Architectures: We plan to investigate other neural network architectures, such as vision transformers, for their potential benefits in satellite imagery analysis. Furthermore, we're keen on exploring the use of other pre-trained models, such as satMAE, which are specifically tailored for satellite imagery processing.
- Advanced Continuity Losses: While the current continuity losses used in our models are effective, we believe there is room for improvement. Future work could involve devising more sophisticated continuity loss functions that offer topological guarantees, ensuring that the resulting river networks are more consistent and accurate.
- Resolution and Generalization Tests: Expanding our models to handle different resolutions of satellite imagery is another important future direction. This will entail conducting rigorous tests on the models' generalization abilities, ensuring they can effectively process and analyze satellite images of varying resolutions.
- Geographical Coverage: While our current focus has been predominantly on Greenland, we aim to test our models on different regions of Greenland, examining their performance across diverse geographical and climatic conditions.
- Climate Science Applications: We plan to collaborate with climate scientists to explore the utility of our SegConnector model in aiding river connection tasks. By comparing our model's performance with legacy methods, we hope to demonstrate its potential in contributing to climate science research.

References

A Appendix

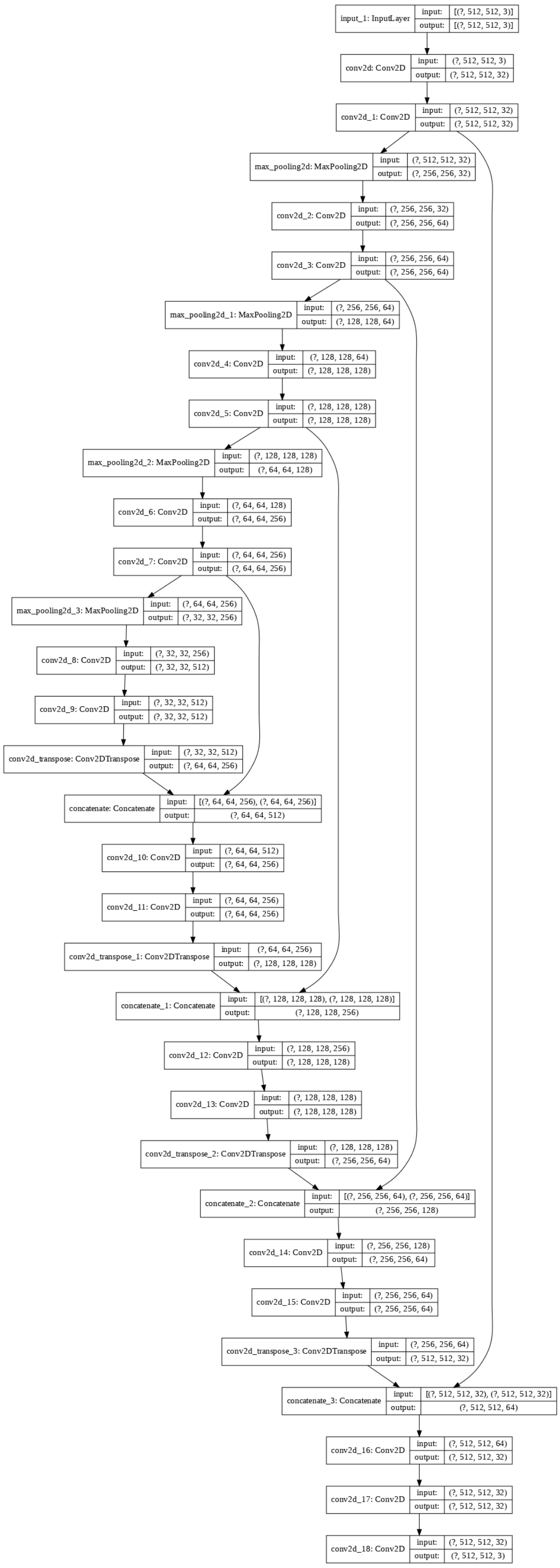

Figure 8: RiverNet UNET architecture

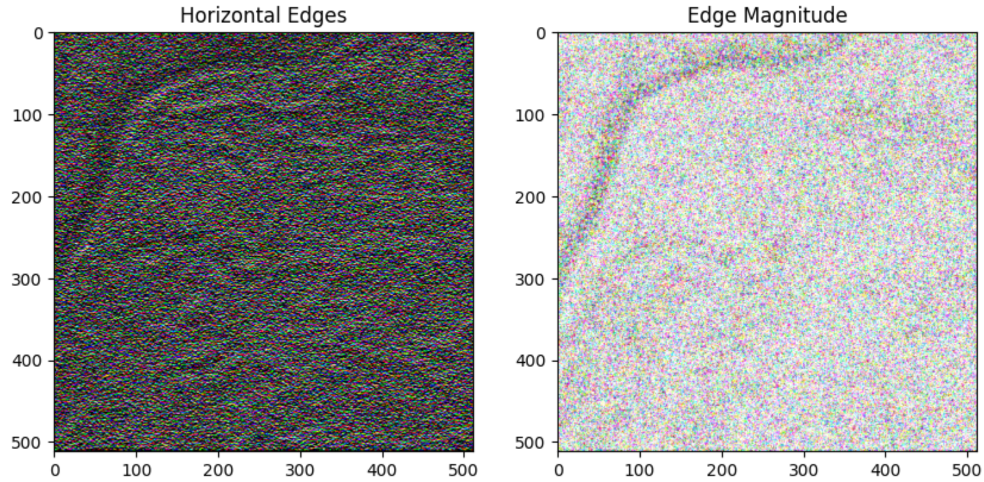

Figure 9: Tests Using Gabor Filter and Edge Detectors as a priori feature extractors

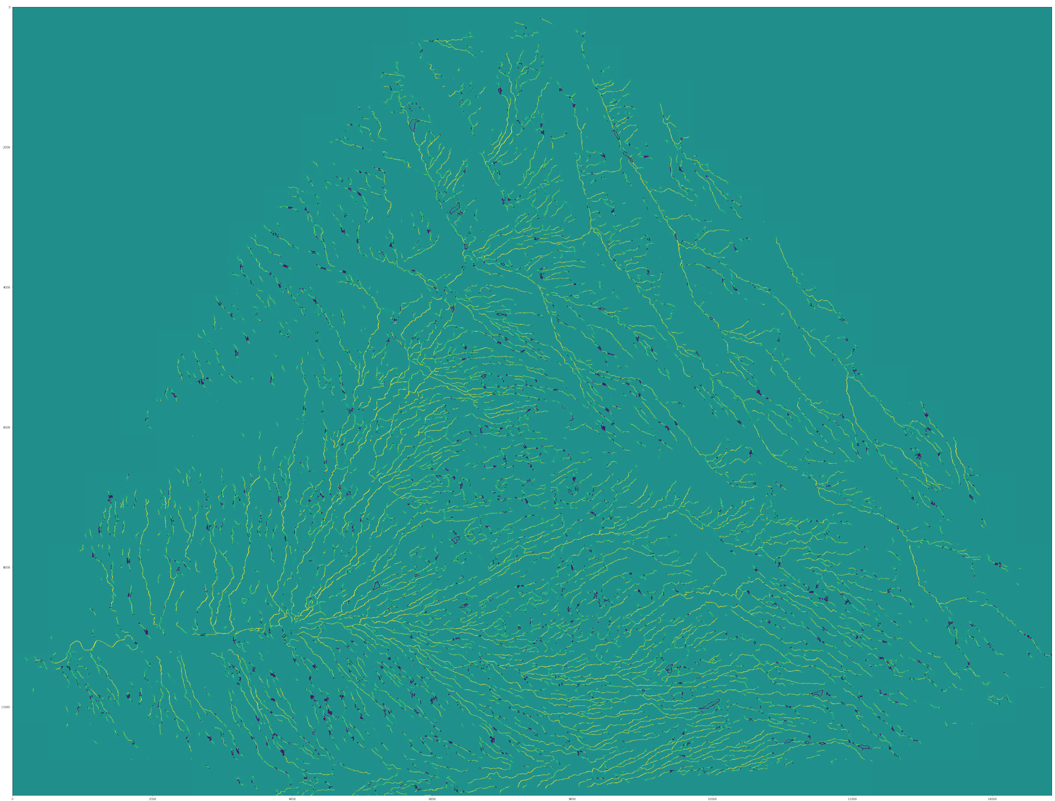

Figure 10: The presence of loops within segmentation predictions

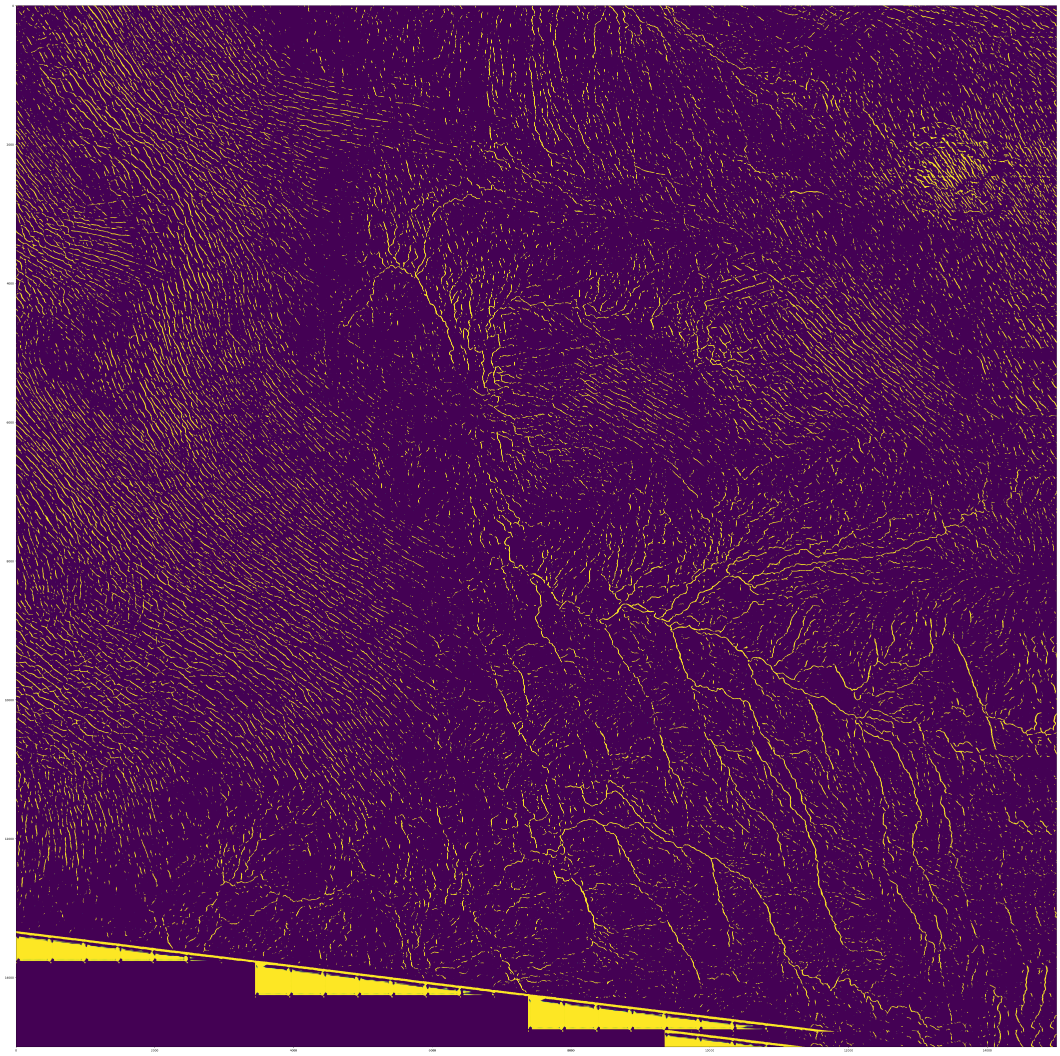

Figure 11: Topological Loss using total variation. Results are inconclusive on the effectiveness of this method