Remote Sensing of Environment 288 (2023) 113495

Contents lists available at ScienceDirect

Remote Sensing of Environment

journal homepage: www.elsevier.com/locate/rse

Mapping retrogressive thaw slumps using deep neural networks

Yili Yang [a,*], Brendan M. Rogers [a], Greg Fiske [a], Jennifer Watts [a], Stefano Potter [a], Tiffany Windholz [a], Andrew Mullen [a], Ingmar Nitze [b], Susan M. Natali [a]

[a] Woodwell Climate Research Center, 149 Woods Hole Road, Falmouth 02540-1644, MA, United States
[b] Alfred Wegener Institute, Telegrafenberg A45, 14473 Potsdam, Germany

ARTICLE INFO

Edited by Jing M. Chen

Keywords:
Retrogressive thaw slumps
Global warming
Semantic segmentation
Remote sensing
Deep learning
Arctic

ABSTRACT

Retrogressive thaw slumps (RTS) are thermokarst features in ice-rich hillslope permafrost terrain, and their occurrence in the warming Arctic is increasingly frequent and has caused dynamic changes to the landscape. RTS can significantly impact permafrost stability and generate substantial carbon emissions. Understanding the spatial and temporal distribution of RTS is a critical step to understanding and modelling greenhouse gas emissions from permafrost thaw. Mapping RTS using conventional Earth observation approaches is challenging due to the highly dynamic nature and often small scale of RTS in the Arctic. In this study, we trained deep neural network models to map RTS across several landscapes in Siberia and Canada. Convolutional neural networks were trained with 965 RTS features, where 509 were from the Yamal and Gydan peninsulas in Siberia, and 456 from six other pan-Arctic regions including Canada and Northeastern Siberia. We further tested the impact of negative data on the model performance. We used 4-m Maxar commercial imagery as the base map, 10-m NDVI derived from Sentinel-2 and 2-m elevation data from the ArcticDEM as model inputs and applied image augmentation techniques to enhance training. The best-performing model reached a validation Intersection over Union (IoU) score of 0.74 and a test IoU score of 0.71. Compared to past efforts to map RTS features, this represents one of the best-performing models and generalises well for mapping RTS in different permafrost regions, representing a critical step towards pan-Arctic deployment. The predicted RTS matched very well with the ground truth labels visually. We also tested how model performance varied across different regional contexts. The result shows an overall positive impact on the model performance when data from different regions were incorporated into the training. We propose this method as an effective, accurate and computationally undemanding approach for RTS mapping.

1. Introduction

Permafrost thaw from a rapidly warming Arctic (Chylek et al., 2022) is projected to generate globally-significant levels of greenhouse gas emissions by the end of the century (Schuur et al., 2015; Gasser et al., 2018; Natali et al., 2021). Thermokarst is an abrupt permafrost thaw process whereby an ice-rich land surface collapses from melting ground ice. Among the terrain-altering changes due to abrupt permafrost thaw events, some of the most rapid and dramatic changes are retrogressive thaw slumps (RTS) (Bernhard et al., 2020). RTS are slope failures that develop on ice-rich hillslope terrain and can progress laterally with headwall retreat rates of up to 40 m (Swanson and Nolan, 2018) or higher per year. RTS features are commonly developed in fine-grained marine deposits of periglacial environments (Slaymaker and Catto, 2017; French, 2017) with high ground ice content vulnerable to melt, water erosion, or surface disturbations such as thermo-erosional niching or active layer detachment (Lafrenière and Lamoureux, 2019; Balser et al., 2014; Kokelj et al., 2017; French, 2017). In recent years, RTS have drawn considerable attention from the scientific community (e.g. Lantz and Kokelj, 2008; Jones et al., 2019; St. Pierre et al., 2018; Bröder et al., 2021; Turetsky et al., 2020) due to their drastic changes on Arctic landscapes and potential carbon emissions. However, the impact of RTS on Arctic carbon-climate feedback is poorly understood, in large part because the spatial distribution and temporal progression of RTS features across large scales are not well characterised. As the global temperature warms, RTS will become an increasingly significant factor in carbon cycling. Therefore it is crucial to develop techniques to robustly map RTS features in different permafrost terrain and regions using high-resolution satellite imagery.

RTS features were previously observed across the Arctic (e.g. Siberia,

* Corresponding author.
E-mail address: yyang@woodwellclimate.org (Y. Yang).

https://doi.org/10.1016/j.rse.2023.113495
Received 6 September 2022; Received in revised form 16 December 2022; Accepted 28 January 2023
Available online 10 February 2023

Alaska and the Canadian Arctic) and on the Tibetan Plateau (e.g. Kokelj et al., 2017; Nitze et al., 2021; Huang et al., 2020b). RTS detection can use methods ranging from satellite remote sensing to airborne photogrammetry and field observation. Although various methods have been used to map RTS features, many depend on manual or semi-automated digitising methods (e.g. Swanson and Nolan, 2018; Jones et al., 2019; Kokelj et al., 2017; Lantz and Kokelj, 2008; Brooker et al., 2014). Lantz and Kokelj (2008) mapped approximately 1100 km^2 of terrain in the Mackenzie Delta region of Canada using aerial photographs. A total of 541 areas affected by thaw slumps were manually digitised. Brooker et al. (2014) mapped RTS in the Richardson Mountains-Peel Plateau region of the Northwest Territories, Canada by thresholding Tasselled Cap values (brightness, greenness and wetness) (Crist and Cicone, 1984) from a Landsat image stack. Bernhard et al. (2020) used two machine learning (ML) algorithms - support vector machine and random forest - on digital elevation model (DEM) difference images derived from the TanDEM-X and Sentinel-2 satellite pairs to map RTS features in northern Canada. Their two models achieved 86% and 87% accuracy in the study region.

Deep learning (DL) has proven to be extremely powerful in computer vision domains such as medical imaging, autopilot and facial recognition (Voulodimos et al., 2018). Mapping RTS with DL methods can greatly facilitate quantitative RTS studies, however, DL approaches have only rarely been applied to mapping RTS using remote sensing. For DL methods, Nitze et al. (2021) used three convolutional neural network (CNN) models - U-Net (Ronneberger et al., 2015), U-Net++ (Zhou et al., 2019) and DeepLabV3 (Chen et al., 2017) - with different encoder backbones on 3-m PlanetScope (PBC, 2018) satellite images. Their best-performing model (U-Net++) achieved an Intersection of Union (IoU) score of 0.58. Huang et al. (2020b) used DeepLabv3+(Chen et al., 2018) on 3-m Planet CubeSat (PBC, 2018) Images to map RTS in the Beiluhe region on the Tibetan Plateau, achieving the highest average precision score (best model for different IoU thresholds) of 0.54. Huang et al. (2022) used a generative adversarial network (GAN) to increase the accuracy of a DeepLabV3+ (Chen et al., 2018) model by generating artificial training data to map RTS in the Southern Arctic and Northern Arctic in Canada. The highest single-region model F1-score reached 0.849. Witharana et al. (2022) used U-Net-based CNN to map RTS in High Arctic Canada, their study provided insights into the impact of image size and model transferability.

For deep learning models, the performance on vision tasks increases logarithmically with training data volume (Sun et al., 2017). Acquiring sufficient training data is usually the bottleneck of training a good DL model. However, there are workarounds for training a DL model with small data sets such as transfer learning, data augmentation and using GAN to generate synthetic training data (e.g. Guo et al., 2020; Wang et al., 2018; Yu et al., 2017; Huang et al., 2022). For our main study area, the Yamal and Gydan peninsulas, the number of existing RTS features is estimated by us to be in the thousands and distributed heterogeneously. In addition, RTS features only cover a tiny percentage (less than 0.01%) of the total area. Therefore, landscape- to regional-scale models that only cover a few hundred RTS features may be prone to under-fitting. However, RTS features can have very different characteristics (size, shape, rate of development) in different Arctic regions (Nitze et al., 2021). It is therefore important to assess whether greater data availability from other Arctic regions improves model performance by providing more examples of within-class variances, or degrades performance by introducing dissimilar RTS characteristics that are beyond the model's capability.

In this work, we aim to develop a computationally lightweight yet accurate DL model to map RTS using high-resolution satellite imagery and explore the potential for extrapolating a regional model to other regions across the Arctic with different geomorphological histories and controls. We put extra effort into making the method lightweight for several reasons. First, we aim to lower the barrier of using DL in RTS studies and make method implementation easier. Second, we avoid the use of costly computing resources or highly customised DL environments in order to facilitate reproducibility and data exchange. We primarily focused on the Yamal and Gydan (YG) peninsulas in Siberia but also introduced six 'extensive sites' from Russia and Canada that were mapped by Nitze et al. (2021) as foreign region RTS data. We used these data sets to address the following questions: for segmentation models with the U-Net paradigm, 1) Do RTS data from foreign regions improve a model developed and applied in a focal region?; and 2) Do RTS data from different regions contribute to a better generalised RTS model applied across all regions? These two questions will shed light on how within-class variance affects an RTS segmentation model. We also aim to understand 3) To what extent adding negative data to the training affects the model performance. This question is essential to address the impact of between-class variance of RTS and their surrounding environments.

2. Materials and methods

2.1. Study sites

We focus on two sets of sites for model development and application. Our primary region is the Yamal and Gydan peninsulas in northwestern Siberia. (Fig. 1 polygon). We then test and apply our model at six previously-mapped 'extensive' sites in various permafrost terrain types affected by RTS across Canada and Russia (Fig. 1 circles).

We used two sets of study sites, the YG site and the extensive sites (see Appendix Table A2. Region details). The Yamal and Gydan peninsulas are located in northwestern Siberia, and collectively cover an area of 412,067 km^2. A total of 509 RTS features were digitised from this region (methods detailed below), including 325 from active RTS and 184 from significantly vegetated or stabilised RTS, and therefore referred to as 'general' RTS features. During the discovery process, 263 RTS-like features were initially recognised as actual RTS. These were marked as non-RTS in the verification stage but were still included in the dataset as counter-examples, and are referred to as negative data. The negative data was used in the training, validation and testing. In this work, 'RTS' is generally used to refer to both active and general RTS features. The six extensive study sites are located in northwest Canada (Bank Island, Herschel Island, Horton Delta) and northwest Siberia (Kolguev Island). These sites are adapted from Nitze et al. (2021)'s work. A total of 456 RTS features were digitised from the extensive sites.

2.2. Satellite imagery data processing

A schematic overview of the complete workflow is shown in Fig. 2. Source imagery for all RTS features included Maxar high-resolution imagery (4-m, Imagery ©2017–2021 Maxar), Sentinel-2 imagery (10-m, 2017–2021, ESA) and the ArcticDEM 2-m elevation data (Porter et al., 2018). The satellite imagery data processing workflow (Fig. 2, Data Processing) was done using Google Earth Engine (GEE) (Gorelick et al., 2017) to process and fuse the multi-source data into a 6-channel composite. The multi-source data represent the components for the three informational aspects of modelling RTS using satellite images: topography, vegetation and elevation. Specifically, we derived a 6-channel composite that included red-green–blue channels (RGB, 1–3 channels), normalised difference vegetation index (NDVI, 4th channel), relative elevation (RE, 5th channel) and enhanced shaded relief (ESR, 6th channel). Processing techniques are further described below.

2.2.1. Sampling

We used the following method to sample the RTS features and rasterise the satellite images to NumPy array objects: for each RTS feature, the centroid coordinates of the RTS location were calculated, and a region-of-interest (ROI) was generated by buffering the centroid with a distance of 0.0025 degrees (around 280 m). We then applied an offset parameter (range from −0.0025 to 0.0025 degrees) to randomly shift

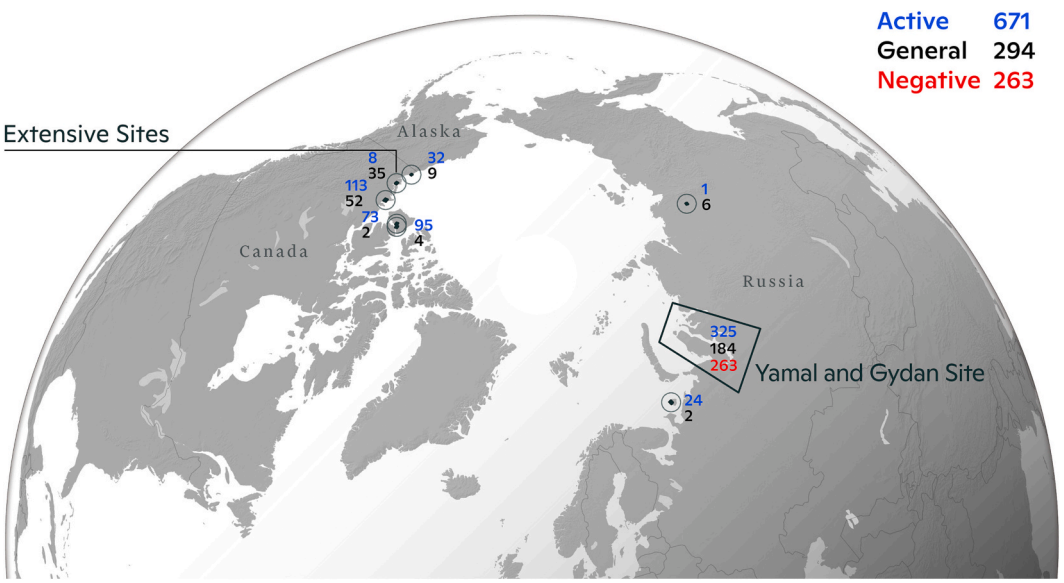

Fig. 1. Map of the study sites. Our primary study site is the Yamal and Gydan peninsulas in northern Siberia. Extensive sites span across the circumpolar domain and are adapted from Nitze et al. (2021). A total of 671 Active RTS features were digitised and labelled in the blue caption. 294 stabilised RTS (referred to as General) were digitised and labelled in the black caption. 263 non-RTS or RTS-like features were digitised and labelled in the red caption.

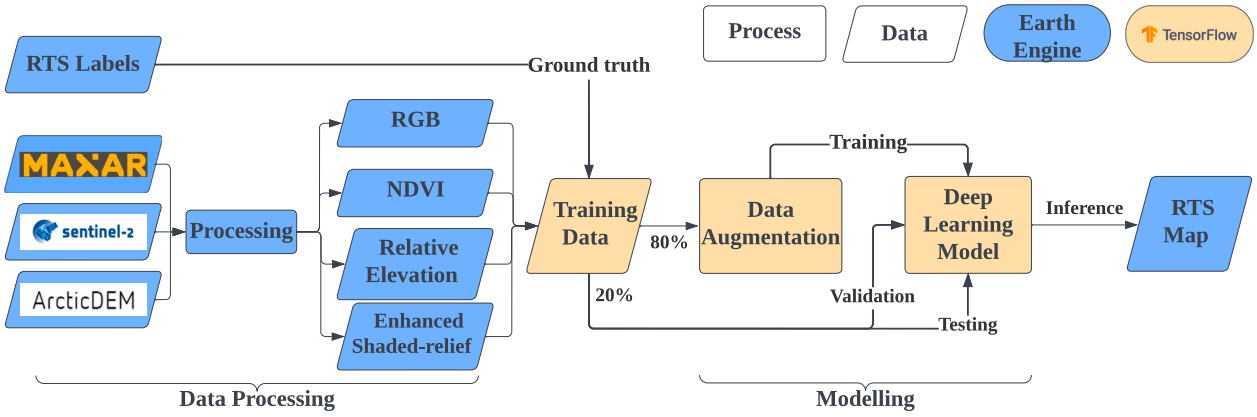

Fig. 2. A schematic workflow of the study. Raw satellite imaging data were processed in Earth Engine into training input data. The training data were split into a training set, a validation set and a testing set using a roughly 8:1:1 ratio. The training set was augmented (Table 1) before input into the DL model. The trained model can be used to make inferences on the whole domain to predict and map undiscovered RTS features.

the ROI, allowing the RTS to be located randomly within the ROI. Without such an adjustment, a model training bias would be introduced by the fact that RTS features would only appear in the centre of an image. The same ROI box was used for all the input and target layers including RGB, NDVI, RE, ESR and the corresponding RTS label. An example of the training images for one RTS site is shown in Fig. 3.

2.2.2. RGB

The colour channels, RGB, are derived from the Maxar data (see Appendix 'Maxar data information'). We acquired 10,551 individual tiles of Maxar Vivid Basic base imagery from 2003 to 2020, of which more than 75% are later than 2015, and composed an image mosaic using GEE. This layer provides information on the visible reflectance of an RTS feature and its surroundings. The resolution of the Maxar data that we purchased is 4-m and it was up-sampled to 2-m to adapt to the ArcticDEM resolution. Values of all three channels were normalised to 0–1. The same upsampling and normalisation were also applied to all other layers described below, resulting in a stack of 2-m resolution inputs.

2.2.3. NDVI

To represent vegetation, we used an NDVI layer, or the normalised difference between the red and the infrared bands, derived from 10-m Sentinel-2 imagery. This layer provides quantification of the green vegetation on and around the RTS site (Fig. 3, NDVI). It is common for an RTS to have lower NDVI values than its surroundings due to the destruction of vegetation through slope failure.

2.2.4. Relative elevation

Relative elevation (RE) was derived from ArcticDEM by subtracting the original image with a mean filter. The RE layer characterizes the difference in elevation between an RTS feature and its surroundings, for example where the elevation is higher due to the headwall of RTS (Fig. 3, Relative Elevation). We tested different values of the kernel size for the mean filter and chose a value of 15 (pixels) as the best parameter for visual comparisons. There are two important considerations for using relative elevation instead of the direct use of absolute elevation values from the DEM. First, after a systematic evaluation of the ArcticDEM quality, we noticed that the DEM quality is often compromised due to interpolated pixel values, null values, artefacts and noise. By using RE,

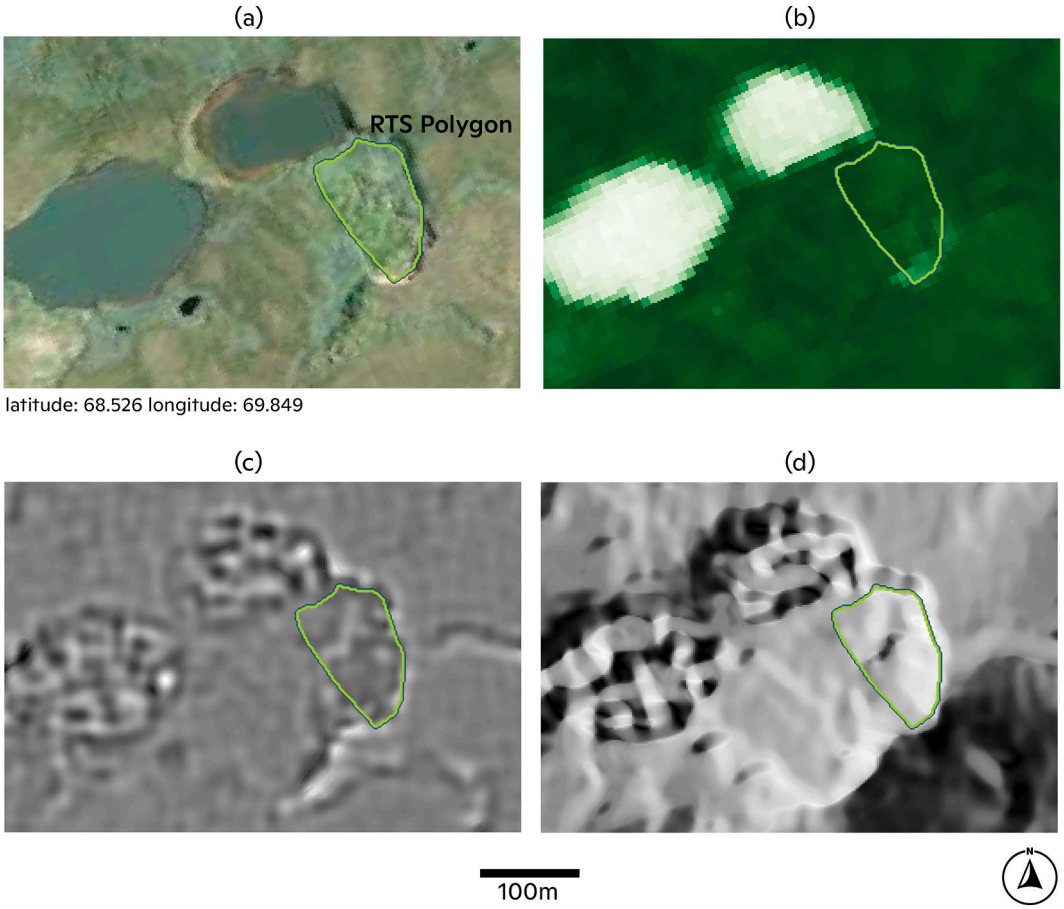

Fig. 3. Visualisation of the input image layers for an example of active RTS feature. (a) The first layer (Maxar) has three visible light channels: red, green and blue showing the colour and outline of the RTS. (b) The second layer (NDVI) emphasises the vegetation characteristics of the RTS. (c) The third layer (RE) emphasises the elevation characteristics of the RTS. (d) The fourth layer (ESR) emphasises the slope and curvature of the RTS. The green polygon overlay on each figure is the manually delineated RTS label.

many of the quality issues are alleviated because the RE layer only focuses on relative elevation changes. Second, we consider RE as a normalisation process that removes the undulation of the terrain while enhancing the signal of an RTS headwall.

2.2.5. Enhanced shaded-relief

The enhanced shaded-relief (ESR) layer was derived from ArcticDEM and Maxar and highlights the slope and headwall of an RTS (Fig. 3, Enhanced Shaded Relief). The ESR layer is produced using the following three steps: First, we computed a multi-directional hillshade from the ArcticDEM using the weighted sum of hillshades from the eight directions (N=0.1, W=0, S=0.1, E=0.3, NE=0, SE=0.1, SW=0.3, NW=0.1). For each direction, the illumination was defined by azimuth and altitude. Different weights, azimuth and altitude combinations were tested and an optimal combination was decided based on visual comparison. Secondly, we used weighted matrix addition with hillshade and slope calculated from the ArcticDEM to produce a weighted shaded relief. Different weights of the slope and hillshade were tested, and we found equal weights yielded the best result. Third, the weighted shaded relief was alpha-blended with the grey scale Maxar data, which enhanced the contrast of an RTS feature from its surroundings. Although the ESR layer and the RE layer both use elevation information from ArcticDEM, they serve different purposes. The RE layer represents the elevation difference between an RTS and its surroundings. The ESR layer represents indirect information derived from elevation i.e. slope and hillshade. Another important consideration for the ESR layer is alpha-blending DEM data with the Maxar data to alleviate mismatch between different sources of satellite imageries by creating a 'fuzziness' on where a discrepancy exists.

2.2.6. RTS labelling

RTS features were manually identified within the imagery and digitised into shapefile polygons using Esri's ArcGIS Pro software. The features served as ground truth data for the model training. High-resolution Esri base map imagery was used as a reference overlay during the RTS delineation process. Every RTS feature was examined at least twice by two trained experts to ensure the RTS features were genuine. If questionable RTS features were found, they were either thrown out or inspected closely using an alternative source of imagery, such as Google base imagery. In some cases, we requested further input from regional experts.

2.2.7. Multi-source satellite data fusion

The use of multi-source satellite data on RTS mapping has an inherent limitation of mismatched features due to the rapid development of RTS. The date of satellite images from different sources can range from a few months to a few years. For the satellite image data source we used, the Sentinel-2 data were from the summer scenes (June-September) between 2017 to 2021. The Maxar images were acquired between 2017 to 2021 for 95% of the 10551 image tiles. The ArcticDEM images we used represent mosaics derived from the strip data files and have an acquisition range of 2009 to 2017. Our approach considered this limitation and mitigated it in two ways:

First, we categorised all the training data into three tiers based on

their degree of mismatch. Only 20% of the training data have well-matched (overlapped boundaries) RTS features between the base map, NDVI and elevation. About 35% of the training data had an intermediate mismatch (boundaries partially overlapped or offset by a few pixels), and 45% of the training data have a significant mismatch (boundaries not overlapped or significantly offset). We then used an oversampling approach by applying weights to the data of different quality so that higher-quality data were trained on more frequently. This weight was set as 10:5:1, i.e. the first-tier data were trained on two times more than the second-tier data and ten times more than the third-tier data.

Secondly, we applied a label smoothing factor to the ground truth images to turn 'hard labels' into 'soft labels', where the confidence in label values was relaxed. We used a label smoothing value of 0.1, which resulted in changing the background-foreground values from 0,1 to 0.05,0.95 on the RTS labels. This method has been theoretically and experimentally shown to benefit partially mislabelled training datasets (Chen et al., 2020; Müller et al., 2019).

2.3. Deep learning model for semantic segmentation of RTS

2.3.1. Development environment

We used TensorFlow and Keras (Python API) to build the DL model based on its integrated ecosystem with GEE and other Google products. Google Cloud Platform and Google Drive were used to host the training data. We used Google Colaboratory Pro+ (Colab) to script the processing and training workflows. We used the cloud GPU (T4/P100) provided by Colab to train the model. The training dataset, model configurations and hyper-parameters were configured using JSON files. The entire workflow was designed to be transferable and undemanding regarding computing resources.

2.3.2. Data augmentation

Deep convolutional neural networks have proven to be effective in semantic segmentation tasks (e.g. Chen et al., 2017; Ronneberger et al., 2015). However, the training of a powerful model relies heavily on a large amount of training data (Shorten and Khoshgoftaar, 2019). One of the biggest challenges in training an RTS segmentation model is the lack of training data because the discovery of RTS features using remote sensing and manual delineation is time-consuming. Data augmentation is a conventional approach to enlarge the training dataset and alleviate issues related to data shortage. Data augmentation, specifically image augmentation, is done by applying transformation, synthesis or degradation to existing data to inflate the training dataset artificially. The basic assumption is that more information can be extracted from the original dataset through augmentations (Shorten and Khoshgoftaar, 2019).

In this study, we applied six different types of basic augmentations: scaling, flipping, affine transformations, elastic transformations, degradation and dropout (Table 1).

Because the purpose of augmentation is to synthesise more heterogeneous training data, we applied each basic augmentation stochastically to maximise the permutation and combination of the transformations on the augmented images. The transformations were applied to each single training image using a probability-based scheme: first, the probability that an image will not receive any augmentation is 10%, which ensures at least 10% of the input is original. Second, there is a 50% probability for each augmentation to be applied. For augmentations that have multiple effects, there is also a 50% probability to choose one/some of the effects. Last, each transformation has a few controlling parameters for its behaviour or intensiveness, and the parameter values are generated randomly by a uniform distribution within a default or arbitrary range. This scheme drastically increases the number of possible outcome augmentations.

When training a neural network, a dataset is trained for many iterations (epochs). To prevent training the exact same images repetitively, which will lead to overfitting, the images were augmented on-the-fly while being sent into the neural network (Fig. 2 Modelling). This allows different augmentations to be applied to the same dataset throughout the training.

Table 1
Basic types of augmentations and their effects.

Basic augmentation	Effect
Scaling	Resize an image
Flipping	Flip an image horizontally, vertically or both
Affine transformation	Rotate and/or shear an image
Elastic transformation	Distort an image elastically
Image degradation	Sharpen and/or blur an image
Dropout	Randomly erase pixel values of an image

2.3.3. Model architecture

We tested four recently-designed models in this study: U-Net++ (Zhou et al., 2019), U-Net3+ (Huang et al., 2020a), TransU-Net (Chen et al., 2021) and ResU-Net (Diakogiannis et al., 2020) (implemented using Sha (2021) GitHub repository). They all belong to the U-Net family, which is a powerful model architecture that proved very successful in all ranges of computer vision tasks. U-Net was first proposed by Ronneberger et al. (2015) as a convolutional neural network architecture for medical imaging. This architecture has been improved in recent years and widely adapted to various scenarios such as remote sensing imagery (e.g. He et al., 2020; Sun et al., 2020; Abdollahi et al., 2021).

A U-Net architecture has three main components (Fig. 4): the encoders, the decoders, and the bottleneck. They are connected by down-sampling and up-sampling paths to form a U-shaped topology. In addition, the skip connections preserve information that could have been lost after many layers of 'passing'.

The encoder blocks can be convolutional layers, which extract different levels of features from the image pattern. The more levels of encoder blocks, the higher level of features can be abstracted, e.g. from lines and curves to complicated semantic meanings (RTS). The encoder of the U-Net is also no longer limited to convolutional layers, and attempts to bring the attention mechanism (Vaswani et al., 2017) into U-Net are ongoing (e.g. TransU-Net and SwinUNet) (Cao et al., 2021). The bottleneck serves as a constraint that 'compresses' feature representations to contain only useful information to reconstruct the input into a segmentation map. The decoder restores the high-dimensional image to the original image size and produces a segmentation map.

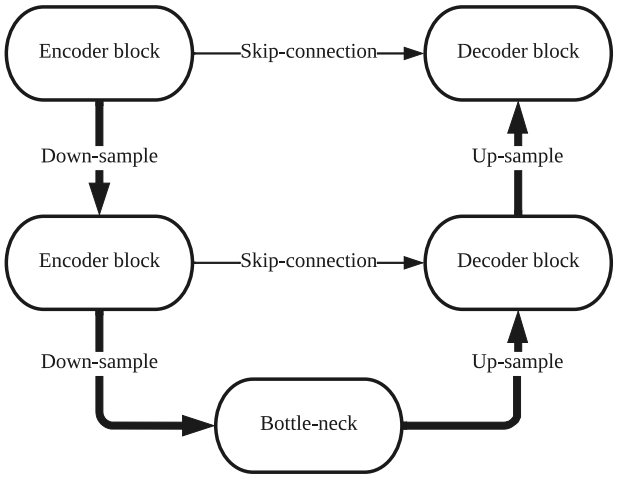

Fig. 4. The abstract topological architecture of U-Net-like models. The model uses an encoder-decoder paradigm, skip-connections connect encoder blocks with decoder blocks by concatenation.

2.3.4. Metrics

We use the Intersection over Union (IoU) score to evaluate the RTS segmentation model accuracy. IoU measures the extent of overlap between predictions and ground truths. An IoU score can range from zero to one, where zero is non-overlap and one represents a perfect overlap.

The IoU metric reveals the model performance on a pixel level, i.e. as a semantic segmentation problem. However, IoU does not indicate the model performance in terms of omission and commission errors when treating each RTS site as an individual object. To fully evaluate the model's performance, we further treated the task as a statistical classification problem and used a confusion matrix to shed light on the accuracy of the model.

We performed a model prediction on the test set and counted numbers of 1) True Positive (TP) when the model prediction identifies a complete RTS, or a portion of an RTS, which can nonetheless help identify the RTS location 2) omitted RTS site as False Negative (FN) and 3) False Positive (FP) when the prediction has zero overlaps with the ground truth. TP, FN and FP were used to calculate the True Positive Rate (TPR/recall), False Discovery Rate (FDR), False Negative Rate (FNR/miss rate), Positive Predictive Value (PPV/precision) and F1 score. Each of the above metrics reveals one aspect of the model's accuracy in terms of detecting RTS features.

2.3.5. Training

Our full dataset contained 965 RTS features that were split into 827 for the training set, 69 for the validation and 69 for the test set. Due to the clustered distribution of RTS in some regions, the sampling method we used can cause partly overlapped training and validation/test images. For this reason, a conventional random split for training-validation-testing can potentially cause data leakage and lead to overestimated accuracy. Therefore our validation and test set were derived manually with RTS locations that are significantly distant from the training RTS locations. The tiling method is defined in Section 2.2.1, the input image dimension is (256,256,6), where the third dimension indicates a 6-band input (R, G, B, NDVI, RE and ESR). Besides the 965 RTS features, we also included the 263 negative data (empty labels) in the training, validation and testing sets.

We trained and tested a total of six models, constituting three different regions with and without the negative dataset. The three different regions are YG, Extensive Sites, and a combination of YG and Extensive Sites. YG is our focal region and the corresponding model represents the baseline of our modelling approach. Extensive Sites were used as 'foreign data' that represent the within-class variance of RTS outside of our focal region. The model trained on the combined region reveals the model's ability to upscale and accept larger within-class variance. Another three models were trained on the three regions using our negative training dataset. The negative dataset represents the highest possible between-class variance and provided the model with counter-examples to learn. The six models were tested on the three regions individually.

Fig. 5 shows the training log of the best-performing model. A total of 185 epochs were trained, which took around 13 h to complete by T4/P100 GPUs on Colab. The length of the training depended on an early stopping criterion to prevent overfitting. We set the stop tolerance to 30 epochs, which resulted in ceasing the training when the validation accuracy did not increase for 30 consecutive epochs. The model parameter on the 30th from the last epoch was saved as the best model. Other stop tolerances from 10 to 50 were tested and 30 was chosen for the best balance between the time allowance to find the best model and training time. The highest IoU achieved on the validation set was 0.74, and the IoU score on the test set was 0.71. Details of the model configuration and training hyperparameters are provided in the appendix.

For tuning the model structure and training hyperparameters, we employed a hybrid tuning method that combines manual, empirical and automatic tuning using Hyperopt (Bergstra et al., 2013), which uses the Tree of Parzen Estimators (Bergstra et al., 2011). Because Hyperopt's tuning results for all the hyperparameters did not result in the best-performing model, only parts of the suggested hyperparameter values were adopted. See Appendix 'Model Parameters and Training Hyper-Parameters' for a full range of tested model configurations and hyperparameters tuning details.

3. Results

We evaluated the four fully trained models on our test data set. The best-performing model was U-Net3+ (0.71), followed by ResU-Net (0.68) and U-Net++ (0.64), where all three models use convolutional layers. The TransU-Net model, which adopts the attention mechanism, performed less well (0.52). The following description of model results will focus on the U-Net3+ model.

Fig. 6 shows six representative examples of the U-Net3+ model prediction on the test set, illustrating both the model's strengths and weaknesses. Overall, the model predictions are visually reliable on both large and small RTS. The predicted RTS boundaries are natural and do not exactly mimic the ground truth labels, which are inherently subjective in many cases, and would otherwise indicate severe overfitting. False positives and false negatives both exist to an acceptable degree (e.g. Fig. 6-c and -f). Most of the RTS headwalls were accurately predicted, and the main discrepancies were located on the edges of the RTS or the thaw slump floors. Lastly, small RTS features can be omitted by the

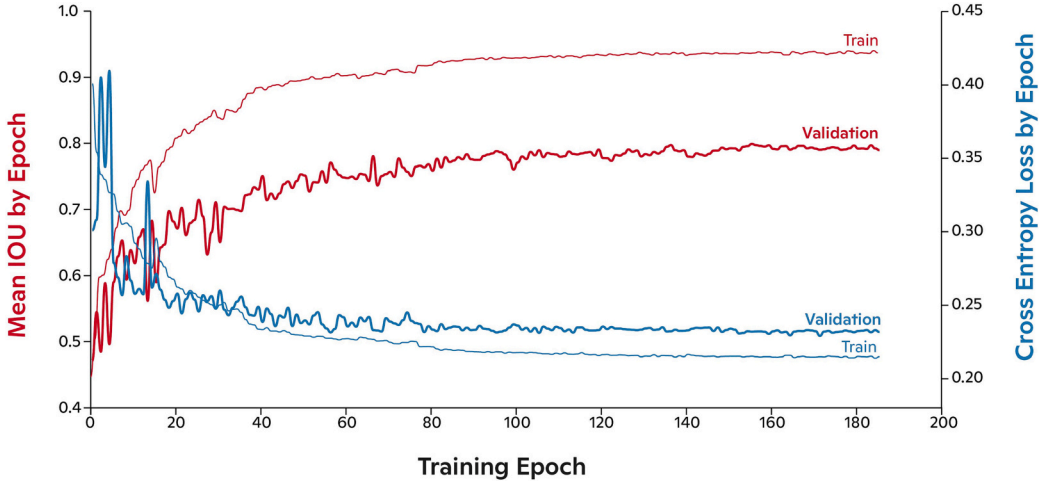

Fig. 5. Loss and accuracy in training and validation by epoch.

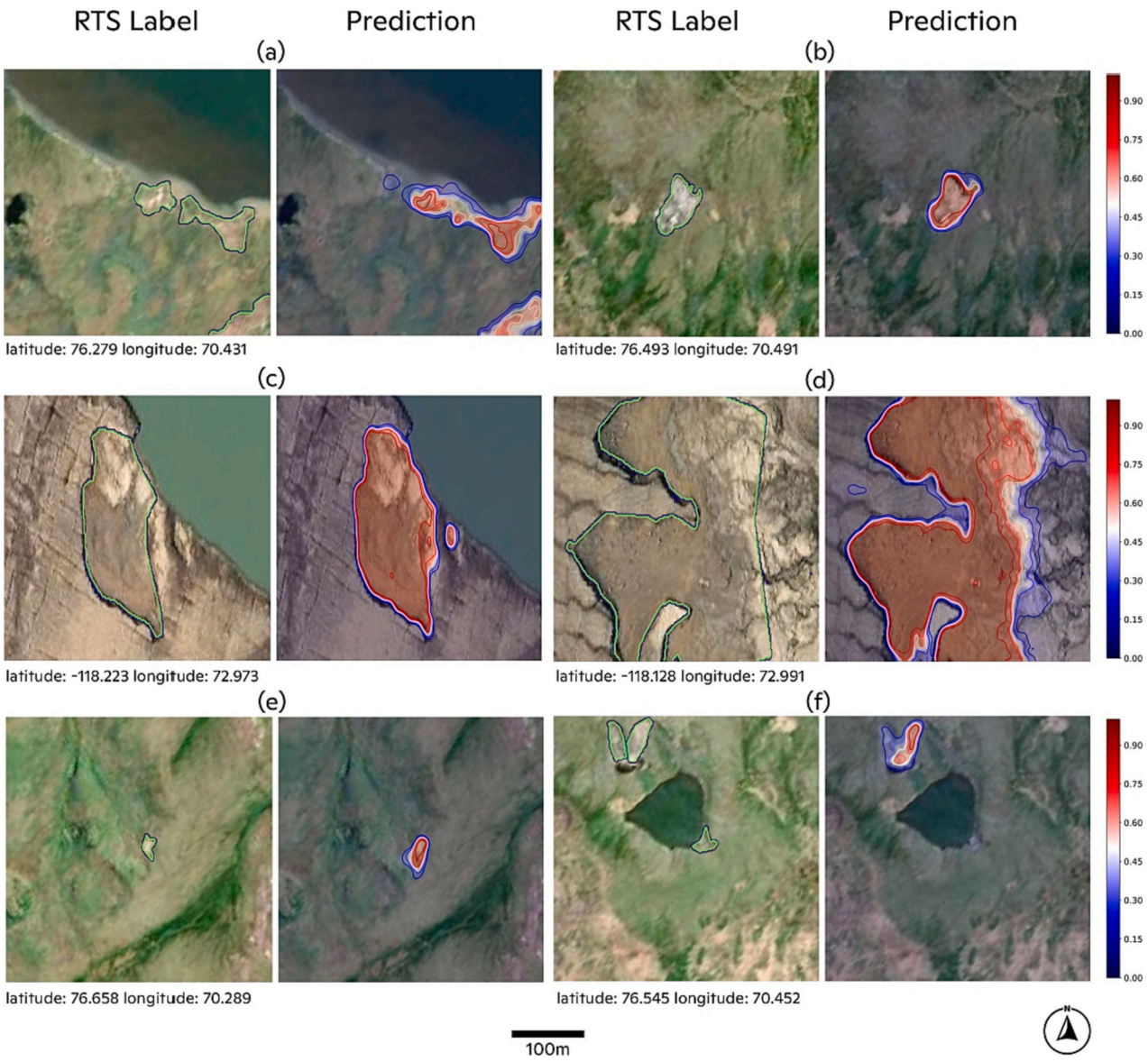

Fig. 6. Examples of RTS segmentation results on the test set. A diverging colour map is used to distinguish prediction probability over and under 0.5. (a) A completely presented RTS cluster adjacent to a waterbody and a partially presented RTS. (b) A well-predicted single, mid-sized inland RTS. (c) A single, large RTS is adjacent to a waterbody with a minor false-positive prediction adjacent to it. (d) A large, connected inland RTS, well-predicted headwalls and ambiguous thaw slump floors. (e) A well-predicted small stabilised RTS. (f) Partially predicted and completely missed RTS.

model (e.g. Fig. 6-f).

The confusion matrix and metrics values are shown in Table 2. The results on the test set show that the model can detect RTS features with an accuracy of 76.79%. The likelihood of detecting false RTS features is 14.85%, and the likelihood of missing a true RTS feature is 23.21%. For all model-detected RTS, 85.15% of them are genuine RTS. F1 score (0.81) is the harmonic mean of precision and recall and gives an overall estimation of the model's reliability. The metrics imply that the model is more prone to omission errors than commission errors. We also performed another model prediction with the negative data in the test set (Table 2). We found that 46 out of 61 false positive predictions resulted from the negative data. All metrics decreased significantly and the F1 score dropped to 0.66. Commission errors became more prevalent when the negative data is added to the testing set.

Fig. 7 shows the model prediction results on the negative data. Fig. 7-a shows an example of a false positive prediction where a high-confident RTS prediction is located on an expert-verified non-RTS feature, which used to be mistakenly recognised as an RTS feature by humans. Fig. 7-b

Table 2
Confusion matrix on the test set and accuracy metrics.

Metrics	Note	Without Negative Data	With Negative data
True Positive (TP)	number of correctly predicted RTS	86	86
False Positive (FP)	number of wrongly predicted RTS	15	61
False Negative (FN)	number of missed RTS	26	26
True Positive Rate (TPR)	$\frac{TP}{TP+FN}$	0.7679	0.7679
False Discovery Rate (FDR)	$\frac{FP}{TP+FP}$	0.1485	0.4150
False Negative Rate (FNR)	$\frac{FN}{TP+FN}$	0.2321	0.2321
Positive Predictive Value (PPV)	$\frac{TP}{TP+FP}$	0.8515	0.5850
F1	$2^{*}\frac{PPV^{*}TPR}{PPV+TPR}$	0.8075	0.6641

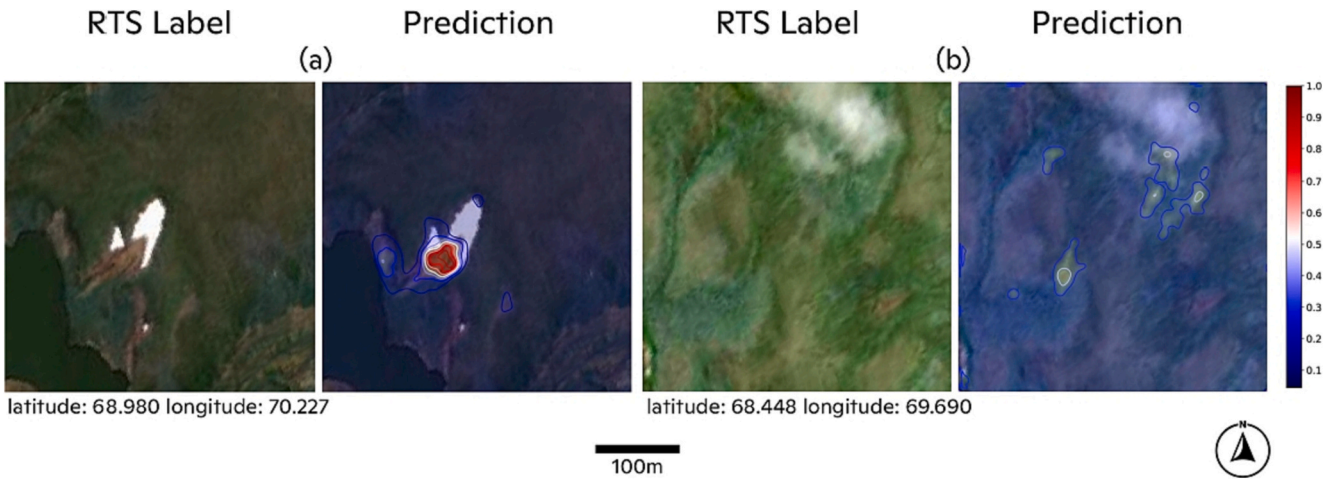

Fig. 7. Examples of model prediction results on the negative dataset. (a) A false-positive RTS feature is predicted on the negative data. (b) A true-negative non-RTS area.

shows a true negative prediction on the negative data where no RTS feature is presented.

The model testing result is shown in Table 3. The results show that: 1) for a particular region or set of regions, adding training data from foreign regions does not affect the model prediction accuracy significantly in the target region; 2) there is a consistent accuracy drop when applying a model developed in one region to foreign regions, indicating insufficient model generalisation; 3) a multi-regional model benefits from more data from different regions, and 4) the YG region has overall lower scores than the Extensive Sites. This is consistent with our observation that RTS in the YG region are generally smaller and harder to recognise compared to other previously-studied regions across the Arctic. 5) Results on the negative data show a consistent minor drop in almost all testing IoU scores.

4. Discussion

We trained a well-performed, multi-regional U-Net3+ model to map RTS features in the Yamal and Gydan peninsulas and other six Arctic regions. We established a DL training workflow to map RTS features using remote sensing data. This workflow is effective, accurate and computationally undemanding and can be conveniently adapted to other computer vision tasks using remote sensing data.

To compare our model with existing models, our multi-region U-Net3+ model (highest IoU=0.76) significantly out-performed Nitze et al. (2021)'s multi-region U-Net++ model (highest IoU=0.58) on the same mapping regions in terms of semantic segmentation. Nitze et al. (2021) chose six different regions in Arctic Canada and Russia to represent a broad variety of environmental conditions and geographic settings, which we adapted as our extensive sites. They used a regional cross-validation training scheme, and the F-score of their model (F-score=0.73) has less detection accuracy than our model (F1=0.81). Our model predictability dropped to F1=0.66 when strongly disturbing RTS-like features from the negative data were included in the test set. Another comparable model is Huang et al. (2022)'s DeepLabV3+ model. Their model is trained with 621 RTS polygons from three regions in Canada's Arctic using PlanetScope imagery. Their F1 score ranged from 0.676 to 0.849 in the three study areas, which fluctuates around our F1 score (0.81). Their highest pixel-wise IoU (pIoU) reached 0.844 on the validation set. However, their model performance on the test set can range from 0.1–0.8, which is much less robust than our model (0.64–0.76). Nevertheless, comparing model performance across regions is challenging due to the heterogeneous nature of RTS feature characteristics and training data.

Table 3
IoU scores of RTS segmentation models trained and tested in different regions (YG, Extensive sites and the combination of both), experimented with negative data included and excluded in both training and testing.

Model Training Region	Model Testing Region					
	YG		Extensive Sites		Combined	
YG	0.65	0.64*	0.52	0.51*	0.57	0.55*
Extensive Sites	0.57	0.54*	0.76	0.73*	0.69	0.68*
Combined	0.63	0.68*	0.76	0.65*	0.71	0.67*

Note: IoU scores without asterisk represent the models trained and tested without negative data. IoU scores with asterisks represent models trained and tested with negative data.

One of the main challenges in using deep learning approaches to map RTS is building a training dataset, including acquiring a satellite base map with good RTS coverage and creating RTS training labels. Manual delineation of RTS features is time-consuming in three aspects. 1) It requires adequate training and experience to identify and label RTS features using satellite imagery, making crowd-sourcing to develop a large-scale RTS training dataset more challenging compared to other DL computer vision projects. 2) The spatial distribution of RTS is sparse and therefore requires time for searching. 3) Drawing RTS polygons manually is time-consuming due to their highly irregular, even fractal boundaries. Hundreds of edges are required for an accurately defined RTS polygon. Often RTS boundaries can be ambiguous due to limited image resolution, which both increases the time for delineation as well as uncertainty. These factors limit the development of an RTS training dataset. The RTS datasets used in previous studies (e.g. Nitze et al., 2021; Huang et al., 2020b) are on the order of a few hundred RTS polygons, yet training models with an under-sized dataset can easily lead to poor generalisation and underfitting. Although data augmentation techniques can alleviate the data shortage, the fundamental solution to this problem is enlarging RTS training datasets from a wider range of Arctic regions. Data sharing across different research groups can accelerate the dataset development process. Additional field data would also help validate and constrain RTS training data derived from remote sensing.

We also identified critical inconsistencies when delineating RTS features. The inconsistencies include 1) whether only the active part of the RTS is delineated or a more general RTS-affected area is included; 2) whether to include or exclude the headwall in the polygon, and 3) how to delineate the ambiguous area where RTS is ending. When these issues are encountered, a common practice is making arbitrary decisions because there is no universally-accepted standard and protocol to follow. These inconsistencies will bring at least two problems,

inaccurate delineation and contradicting datasets.

The use of negative data produced a surprising result that counter-examples have a minor negative impact on the model performance, the cause of which is not fully understood. We expected an increase in model performance by adding negative data because it introduces more examples of between-class variation to the model, therefore, increasing its ability to separate. It is worth stressing that the negative data arose from confusing features on the ground that were originally identified as RTS but later rejected, hence they are more challenging features than the average background. One explanation could be that the scale of the current model does not provide the capability to handle highly ambiguous features that are even difficult for humans to differentiate. In this case, the result implies that the RTS-like negative data is a disturbance to the model.

Our experimental results provide important implications for the use of deep learning to map RTS using satellite imagery. First, the within-class variance of RTS is large, and therefore a model trained on one region may not represent RTS well in other regions, as shown by Nitze et al. (2021). Data from foreign regions also does not improve the model performance for a given region. However, when mapping RTS in a set of regions, adding more data from the subset of regions can benefit the model performance i.e. the within-class variance of RTS can be overcome by training a multi-regional model. This supports the hypothesis that it is possible to build a general model that can account for all RTS variances across the Arctic. Second, the between-class variance of RTS is small, as our current model does not have the capability to outperform humans in differentiating highly ambiguous non-RTS features from genuine RTS features. More expansive testing is needed to confirm our approach across other or larger regions.

4.1. Future research directions

Based on our findings, we suggest the following for future research priorities: 1) Using our model formulation, wall-to-wall maps and spatial analysis of RTS features across the Yamal and Gydan Peninsulas can be generated. 2) Building a common, open-source, pan-Arctic RTS training dataset would be highly beneficial for a variety of RTS-related research including mapping, analysis, and process modelling. 3) Developing an agreed-upon RTS labelling protocol would unify dataset standards and facilitate data sharing. 4) We suggest training and testing models with more neutral negative data where there are no confusing RTS features to further understand the impact of negative data as model inputs. 5) More extensive and systematic studies can explore the relative differences, mechanisms, and model performances between deep learning model architectures when mapping RTS features. 6) We suggest training a similar model with 3-m Planet imagery, which has a different cost structure and may be feasible for scaling to large areas. Finally, we suggest employing our modelling approach across different regions in the Arctic, prioritising areas where RTS occurred frequently in recent years, such as Banks Island and the Peel Plateau in Canada. This is particularly important considering our favourable model performance compared to past work, and the fact that we focused on one of the most challenging domains to map RTS given their size - the YG peninsulas.

5. Summary

In this study, we aimed to map RTS using deep neural networks. We established a framework for multi-source data fusion, data processing and DL model training with remote sensing that other researchers can utilise and build from. We trained multi-regional semantic segmentation models using U-Nets to map RTS in selected circumpolar regions. We tested four different U-Net-like models and the best-performing model (U-Net3+) reached an IoU score of 0.71 on the test set, which outperformed most of the published deep learning models created for this task. We also tested the impact of training on different regions and training with negative data, and the results show a positive impact of enlarging the training dataset on the multi-regional model. Adding foreign training data to a single-region model has a neutral impact. Training with counter-examples has a minor negative impact on the deep learning model performance. We believe that the bottleneck for the current model is increasing the size of the training set. The overall result is encouraging for a more generalised RTS segmentation model that accounts for different regional contexts of RTS in the pan-Arctic. Because we expect many more RTS events to occur in the near future due to climate warming, it is crucial to have a model approach capable of detecting and quantifying these features so we can be prepared to track these rapid disturbances. Consistently mapping RTS across the Arctic would improve the scientific community's ability to document and understand rapid changes, and to project permafrost carbon feedback under various climate scenarios.

CRediT authorship contribution statement

Yili Yang: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Brendan M. Rogers:** Conceptualization, Resources, Writing – review & editing, Supervision, Funding acquisition. **Greg Fiske:** Funding acquisition, Conceptualization, Writing – review & editing, Data curation, Visualization. **Jennifer Watts:** Writing – review & editing. **Stefano Potter:** Writing – review & editing. **Tiffany Windholz:** Data curation. **Andrew Mullen:** Writing – review & editing. **Ingmar Nitze:** Resources, Writing – review & editing. **Susan M. Natali:** Conceptualization, Supervision, Writing – review & editing, Funding acquisition, Resources.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yili Yang reports financial support was provided by Heising-Simons Foundation.

Data availability

Data will be made available on request.

Acknowledgements

This research was funded by the Heising Simons Foundation [Grant No. 2021–3040]. We acknowledge support from Faculty AI at the initial stage of this research. This project was also supported by the HGF AI-CORE project. Additional funding was provided by the NSF Permafrost Discovery Gateway projects (NSF Grants #2052107 and #1927872).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.rse.2023.113495.

References

Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., Alamri, A., 2021. Multi-object segmentation in complex urban scenes from high-resolution remote sensing data. Remote Sens. 13 (18), 3710.

Balser, A.W., Jones, J.B., Gens, R., 2014. Timing of retrogressive thaw slump initiation in the noatak basin, northwest alaska, usa. J. Geophys. Res.: Earth Surf. 119 (5), 1106–1120.

Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B., 2011. Algorithms for hyper-parameter optimization. Adv. Neural Inf. Process. Syst. 24.

Bergstra, J., Yamins, D., Cox, D., 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In: International conference on machine learning. PMLR, pp. 115–123.

Bernhard, P., Zwieback, S., Leinss, S., Hajnsek, I., 2020. Mapping retrogressive thaw slumps using single-pass tandem-x observations. IEEE J. Select. Top. Appl. Earth Obs. Remote Sens. 13, 3263–3280.

Bröder, L., Keskitalo, K., Zolkos, S., Shakil, S., Tank, S.E., Kokelj, S.V., Tesi, T., Van Dongen, B.E., Haghipour, N., Eglinton, T.I., et al., 2021. Preferential export of permafrost-derived organic matter as retrogressive thaw slumping intensifies. Environ. Res. Lett. 16 (5), 054059.

Brooker, A., Fraser, R.H., Olthof, I., Kokelj, S.V., Lacelle, D., 2014. Mapping the activity and evolution of retrogressive thaw slumps by tasselled cap trend analysis of a landsat satellite image stack. Permafrost Periglac. Process. 25 (4), 243–256.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M., 2021. Swin-unet: Unet-like pure transformer for medical image segmentation. arXiv preprint arXiv: 2105.05537.

Chen, B., Ziyin, L., Wang, Z., Liang, P.P., 2020. An investigation of how label smoothing affects generalization. arXiv preprint arXiv:2010.12648.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021. Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306.

Chen, L.-C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818.

Chylek, P., Folland, C., Klett, J.D., Wang, M., Hengartner, N., Lesins, G., Dubey, M.K., 2022. Annual mean arctic amplification 1970–2020: Observed and simulated by cmip6 climate models. Geophys. Res. Lett., e2022GL099371

Crist, E.P., Cicone, R.C., 1984. A physically-based transformation of thematic mapper data—the tm tasseled cap. IEEE Trans. Geosci. Remote Sens. 3, 256–263.

Diakogiannis, F.I., Waldner, F., Caccetta, P., Wu, C., 2020. Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data. ISPRS J. Photogramm. Remote Sens. 162, 94–114.

French, H.M., 2017. The periglacial environment. John Wiley & Sons.

Gasser, T., Kechiar, M., Ciais, P., Burke, E., Kleinen, T., Zhu, D., Huang, Y., Ekici, A., Obersteiner, M., 2018. Path-dependent reductions in co2 emission budgets caused by permafrost carbon release. Nat. Geosci. 11 (11), 830–835.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R., 2017. Google earth engine: Planetary-scale geospatial analysis for everyone. Remote Sens. Environ. 202, 18–27.

Guo, X., Chen, Y., Liu, X., Zhao, Y., 2020. Extraction of snow cover from high-resolution remote sensing imagery using deep learning on a small dataset. Remote Sens. Lett. 11 (1), 66–75.

He, N., Fang, L., Plaza, A., 2020. Hybrid first and second order attention unet for building segmentation in remote sensing images. Sci. China Inf. Sci. 63 (4), 1–12.

Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.-W., Wu, J., 2020a. Unet 3+: A full-scale connected unet for medical image segmentation. In: ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 1055–1059.

Huang, L., Lantz, T.C., Fraser, R.H., Tiampo, K.F., Willis, M.J., Schaefer, K., 2022. Accuracy, efficiency, and transferability of a deep learning model for mapping retrogressive thaw slumps across the canadian arctic. Remote Sens. 14 (12), 2747.

Huang, L., Luo, J., Lin, Z., Niu, F., Liu, L., 2020b. Using deep learning to map retrogressive thaw slumps in the beiluhe region (tibetan plateau) from cubesat images. Remote Sens. Environ. 237, 111534.

Jones, M.K.W., Pollard, W.H., Jones, B.M., 2019. Rapid initialization of retrogressive thaw slumps in the canadian high arctic and their response to climate and terrain factors. Environ. Res. Lett. 14 (5), 055006.

Kokelj, S.V., Lantz, T.C., Tunnicliffe, J., Segal, R., Lacelle, D., 2017. Climate-driven thaw of permafrost preserved glacial landscapes, northwestern canada. Geology 45 (4), 371–374.

Lafrenière, M.J., Lamoureux, S.F., 2019. Effects of changing permafrost conditions on hydrological processes and fluvial fluxes. Earth Sci. Rev. 191, 212–223.

Lantz, T.C., Kokelj, S.V., 2008. Increasing rates of retrogressive thaw slump activity in the mackenzie delta region, nwt, canada. Geophys. Res. Lett. 35 (6).

Müller, R., Kornblith, S., Hinton, G.E., 2019. When does label smoothing help? Adv. Neural Inf. Process. Syst. 32.

Natali, S.M., Holdren, J.P., Rogers, B.M., Treharne, R., Duffy, P.B., Pomerance, R., MacDonald, E., 2021. Permafrost carbon feedbacks threaten global climate goals. Proc. Nat. Acad. Sci. 118 (21), e2100163118.

Nitze, I., Heidler, K., Barth, S., Grosse, G., 2021. Developing and testing a deep learning approach for mapping retrogressive thaw slumps. Remote Sens. 13 (21), 4294.

PBC, P.L., 2018–. Planet application program interface: In space for life on earth. URL: https://api.planet.com.

Porter, C., Morin, P., Howat, I., Noh, M., Bates, B., Peterman, K., Keesey, S., Schlenk, M., Gardiner, J., Tomko, K., et al., 2018. Arcticdem 2018. https://doi. org/10.7910/DVN/OHHUKH 413.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer, pp. 234–241.

Schuur, E.A., McGuire, A.D., Schädel, C., Grosse, G., Harden, J.W., Hayes, D.J., Hugelius, G., Koven, C.D., Kuhry, P., Lawrence, D.M., et al., 2015. Climate change and the permafrost carbon feedback. Nature 520 (7546), 171–179.

Sha, Y., 2021. Keras-unet-collection.https://github.com/yingkaisha/keras-unet-collection.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. J. Big Data 6 (1), 1–48.

Slaymaker, O., Catto, N., 2017. Landscapes and landforms of Western Canada. Springer.

St. Pierre, K.A., Zolkos, S., Shakil, S., Tank, S.E., St. Louis, V.L., Kokelj, S.V., 2018. Unprecedented increases in total and methyl mercury concentrations downstream of retrogressive thaw slumps in the western canadian arctic. Environ. Sci. Technol. 52 (24), 14099–14109.

Sun, C., Shrivastava, A., Singh, S., Gupta, A., 2017. Revisiting unreasonable effectiveness of data in deep learning era. In: Proceedings of the IEEE international conference on computer vision, pp. 843–852.

Sun, S., Mu, L., Wang, L., Liu, P., 2020. L-unet: An lstm network for remote sensing image change detection. IEEE Geosci. Remote Sens. Lett.

Swanson, D.K., Nolan, M., 2018. Growth of retrogressive thaw slumps in the noatak valley, alaska, 2010–2016, measured by airborne photogrammetry. Remote Sens. 10 (7), 983.

Turetsky, M.R., Abbott, B.W., Jones, M.C., Anthony, K.W., Olefeldt, D., Schuur, E.A., Grosse, G., Kuhry, P., Hugelius, G., Koven, C., et al., 2020. Carbon release through abrupt permafrost thaw. Nat. Geosci. 13 (2), 138–143.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E., 2018. Deep learning for computer vision: A brief review. Comput. Intell. Neurosci. 2018.

Wang, A.X., Tran, C., Desai, N., Lobell, D., Ermon, S., 2018. Deep transfer learning for crop yield prediction with remote sensing data. In: Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies, pp. 1–5.

Witharana, C., Udawalpola, M.R., Liljedahl, A.K., Jones, M.K.W., Jones, B.M., Hasan, A., Joshi, D., Manos, E., 2022. Automated detection of retrogressive thaw slumps in the high arctic using high-resolution satellite imagery. Remote Sens. 14 (17), 4132.

Yu, X., Wu, X., Luo, C., Ren, P., 2017. Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. GISci. Remote Sens. 54 (5), 741–758.

Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J., 2019. Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE Trans. Med. Imaging 39 (6), 1856–1867.