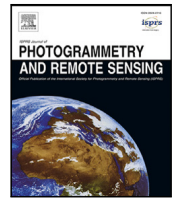




Contents lists available at ScienceDirect

## ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: [www.elsevier.com/locate/isprsjprs](http://www.elsevier.com/locate/isprsjprs)

## Identifying active retrogressive thaw slumps from ArcticDEM

Lingcao Huang<sup>a,\*</sup>, Michael J. Willis<sup>b,c</sup>, Guiye Li<sup>d</sup>, Trevor C. Lantz<sup>e</sup>, Kevin Schaefer<sup>f</sup>, Elizabeth Wig<sup>g</sup>, Guofeng Cao<sup>d</sup>, Kristy F. Tiampo<sup>b</sup><sup>a</sup> Earth Science and Observation Center, Cooperative Institute for Research in Environmental Sciences (CIRES), University of Colorado Boulder, Boulder, CO, USA<sup>b</sup> Geological Sciences and CIRES, University of Colorado Boulder, Boulder, CO, USA<sup>c</sup> Department of Geosciences, Virginia Tech, Blacksburg, VA, USA<sup>d</sup> Department of Geography, University of Colorado Boulder, Boulder, CO, USA<sup>e</sup> School of Environmental Studies, University of Victoria, Victoria, BC, Canada<sup>f</sup> National Snow and Ice Data Center, CIRES, University of Colorado Boulder, Boulder, CO, USA<sup>g</sup> Department of Electrical Engineering, Stanford University, Stanford, Stanford, CA, USA

## ARTICLE INFO

Dataset link: <http://arcticdem.org>, <https://arcticdata.io/>, [https://github.com/yghlc/rs\\_data\\_pr oc](https://github.com/yghlc/rs_data_pr oc), [https://github.com/yghlc/Landuse\\_DL](https://github.com/yghlc/Landuse_DL), <http://github.com/yghlc/labelEarth>, <https://github.com/yghlc/labelEarthServer>

## Keywords:

ArcticDEM  
YOLO  
Crowdsourcing  
Deep learning  
Feature extraction  
Permafrost  
Retrogressive thaw slump

## ABSTRACT

Permafrost degradation in the Arctic is accelerating and affects northern communities, ecosystems, and global soil carbon storage. However, the extent, distribution, and rates of permafrost degradation in the pan-Arctic remain unknown, contributing to the challenges of mapping and monitoring it in a harsh and remote environment. We applied feature extraction, deep learning, and crowdsourcing to an open-access, high-resolution (2 m), and multi-temporal digital elevation models (i.e. ArcticDEM), to identify retrogressive thaw slumps (RTSs), a dynamic form of permafrost degradation widespread across the Arctic. Specifically, we (1) developed an automated pipeline to process approximately 200 TB of ArcticDEM data; (2) designed feature extractors (pixel-wise elevation differences, polygons of elevation reductions, lines of narrow-steep slopes, and RTS headwall lines) to identify slumps; (3) trained a super-efficient object detection algorithm based on deep learning (YOLOv4) and used it to locate RTSs from composite imagery derived from the ArcticDEM; (4) combined the extracted features and the bounding boxes output by YOLOv4 to obtain mapping results at a manageable level; and (5) developed a crowdsourcing system and invited volunteers to validate and refine the results. The final map included 2494 RTSs (actively expanding during ArcticDEM observation) across the Arctic. The results also show that (1) it is necessary to combine the extracted features and deep learning to remove many false positives in the scenario with limited training data, but large regions to map; (2) some hotspots of RTSs need further and detailed investigation, including an RTS cluster in Greenland; (3) the crowdsourcing system is useful for the validation of a large dataset, but responses to this work were limited, possibly because RTSs are a quite specific topic that not many people are familiar with. The results likely miss many RTSs due to the limitation in the method for identifying RTSs and uncertainties as well as short observation periods of ArcticDEM at many locations. This study provides data and serves as a starting point to develop a global inventory and better understand permafrost thaw in the pan-Arctic using very high resolution remote sensing.

## 1. Introduction

Increases in ground temperature across the circumpolar (Biskaborn et al., 2019; Smith et al., 2022), are altering active layer thickness (Åkerman and Johansson, 2008; Pang et al., 2012), causing top-down thaw, and intensifying thermokarst activity (Lewkowicz and Way, 2019). Permafrost degradation is concerning because frozen soils contain a significant amount of carbon (Tarnocai et al., 2009) which may be released to atmosphere under climate change (Zimov et al., 2006; Schuur et al., 2015; Mu et al., 2020; Turetsky et al., 2020; Natali et al., 2021; Miner et al., 2022). Moreover, abrupt thawing can cause hydrologic

and geomorphic changes that impact northern communities and ecosystems (Hjort et al., 2018, 2022; Teufel and Sushama, 2019; Schaefer et al., 2020; Miner et al., 2021). Abrupt thawing can also produce distinct landforms (Jorgenson, 2013) such as thermokarst lakes (e.g., Lin et al., 2010), retrogressive thaw slumps (e.g., Burn and Lewkowicz, 1990), melt ponds (e.g., Steedman et al., 2017), and thermo-erosion gullies (e.g., Godin et al., 2014; Huang et al., 2018). Distributed across the circumpolar, these landforms are challenging to map because they are small and dynamic. Developing comprehensive inventories of these features to better understand the impacts of permafrost degradation on

\* Corresponding author.

E-mail address: [lingcao.huang@colorado.edu](mailto:lingcao.huang@colorado.edu) (L. Huang).<https://doi.org/10.1016/j.isprsjprs.2023.10.008>

Received 21 January 2023; Received in revised form 16 September 2023; Accepted 8 October 2023

Available online 20 October 2023

0924-2716/© 2023 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

the global climate requires innovative methods to rapidly process and analyze high resolution data.

A retrogressive thaw slump (RTS) is a form of abrupt permafrost thaw (e.g. Fig. 1), which consists of: (1) a nearly-vertical headwall that retreats upslope as ground ice in the headwall melts, and (2) a scar area where thawed materials are deposited (Burn and Friele, 1989; Wang et al., 2009; Niu et al., 2016; Lewkowicz and Way, 2019). An RTS typically has a depression with a cusp or horseshoe shaped scar area visible on remote sensing imagery (e.g., Kokelj et al., 2009, 2017; Ward Jones et al., 2019). In some locations, slump scars also can have irregular or elongated shapes (e.g., Huang et al., 2021; Bernhard et al., 2022b). In a well-vegetated region, removal of vegetation by an RTS causes strong spectral differences between the RTS and its surroundings on satellite imagery acquired in visible bands (e.g., Laclelle et al., 2015). In Tibet and the high Arctic, where vegetation cover is low, an RTS is quite similar to its surroundings on satellite imagery (e.g., Xia et al., 2022; Ward Jones et al., 2019). In the past several decades, the area and number of RTSs have significantly increased in several regions including the Mackenzie Delta Region (Lantz and Kokelj, 2008), the Beiluhe Region in Tibet (Luo et al., 2019), Banks Island (Lewkowicz and Way, 2019), and the northern Taymyr Peninsula (Bernhard et al., 2022b), but their spatial distribution for the entire permafrost region is still unknown.

Many studies have used manual or automated methods to identify and delineate RTSs from remote sensing imagery in different regions. Manual digitization is the main approach to compiling RTS inventories in many local regions including Tibet (e.g., Niu et al., 2016; Luo et al., 2019, 2022) and the Arctic (e.g., Ramage et al., 2017; Lewkowicz and Way, 2019), but it is labor intensive and time-consuming. Automated or semi-automated techniques utilizing Landsat image stacks (Brooker et al., 2014; Nitze et al., 2018; Lara et al., 2019), high-resolution satellite imagery (Rudy et al., 2013), and digital elevation models (Bernhard et al., 2020; van der Sluijs et al., 2022) have been explored to map RTSs, but these approaches often miss small features or are typically limited to local or regional sites.

In the past decade, deep learning has significantly improved the state-of-the-art in image processing and is widely used in many applications (LeCun et al., 2015). Deep learning is a machine learning technique built on neural networks and can automatically learn pattern from data (Goodfellow et al., 2016). Before deep learning, people had to manually design feature extractors and obtain features (termed as extracted features hereafter) such as edges and textures from the image itself using various algorithms in image processing and computer vision. These extracted features require people have strong skills and domain knowledge to design feature extractors for specific tasks, while deep learning does not have this requirement. Since Krizhevsky et al. (2012) published a milestone work, deep learning has quickly dominated data processing in many fields. Deep learning approaches have been used to map RTSs, but the transferability of these methods when scaling up from local to regional or continental extents is currently unsatisfactory.

Several recent studies have used deep learning to map RTSs (Huang et al., 2020, 2022; Nitze et al., 2021) and quantify their evolution (Huang and Liu, 2020; Huang et al., 2021). Huang et al. (2020) automatically delineated 220 RTSs in a homogeneous region in Tibet using a trained model, but achieved a slightly lower performance in a new region where training data was not available. Nitze et al. (2021) and Huang et al. (2022) conducted experiments using training data from several regions and found that the transferability of trained models was hard to improve even using data augmentation and generative adversarial networks to increase the volume and diversity of training data. The challenge of transferability is related to the fact that the deep learning approaches used in these studies are based on supervised learning, which requires a significant amount of training data from every possible scenario. A neural network trained with sufficient data should have perfect transferability (Zhang et al., 2021), but it is unclear

how much training data is required to accurately map RTSs at the pan-Arctic scale.

Currently, a lack of training data and very high-resolution & publicly available satellite imagery hinders efforts to build a comprehensive inventory of RTSs in the pan-Arctic. Training data with sufficient volume and diversity is necessary for training a deep learning model to map RTSs in large scales. Many small RTSs are only visible in high-resolution (<5 m) satellite imagery, which is not available to the public or general research communities.

Our objectives are to: (1) generate features that represent temporal and spatial changes of topography in the entire ArcticDEM domain, excepting the Greenland Ice Sheet, (2) locate RTSs (actively expanding during ArcticDEM observation) using a combination of extracted features and deep learning, and (3) develop a crowdsourcing system and invite effort from scientific communities to refine the automated mapping results. Specifically, we use the ArcticDEM (Section 2), an open-access, multi-temporal, and high resolution (2 m) digital elevation models (DEM). We generate extracted features including: polygons of elevation reductions, line features representing abrupt slope changes, and time series of RTS headwall movements. We show that the combination of the extracted features and deep learning is necessary to overcome the challenge of limited training data. Due to the challenges of processing big data and mapping RTSs at a pan-Arctic scale, we focus on the feasibility of this project and try to identify barriers hindering the effort of compiling a complete RTS inventory. We also develop an automated pipeline to process this huge dataset (~200 TB) with limited storage quota (10 TB) on a supercomputer. Our work is the first attempt to inventory RTSs at a pan-Arctic scale and provides an important dataset to advance the monitoring of permafrost thaw across the Arctic.

## 2. ArcticDEM

The ArcticDEM is a digital surface model of the Arctic generated using optical stereo imagery acquired by Worldview-1, 2, 3, and GeoEye-1 satellites (Morin et al., 2016; Porter et al., 2018) and the Surface Extraction from TIN-based Search-space Minimization (SETSM) algorithm (Noh and Howat, 2015, 2017). This study uses Release 7 issued by the Polar Geospatial Center (PGC) in 2018 and the corresponding mosaic and strip products at a spatial resolution of two meters ([arcticdem.org](http://arcticdem.org)) and vertical precision of better than 0.5 m (Morin et al., 2016). The ArcticDEM covers all land area north of 60 °N, and southern Greenland, southern Alaska, and the Kamchatka Peninsula in Russia (Fig. 2a) and has various numbers (1–10+) of observations at different locations (Fig. 2b). The ArcticDEM spans the period from 2008 to 2017 and includes acquisition from all months with the most observations in April (Figs. 3a and b). The ArcticDEM is a digital surface model that represents the height of the Earth's surface including vegetation, snow, and buildings etc. According to the PGC website, the strip files correspond to the overlapping area of stereopair image swaths, with 16–18 km width and 110–120 km length; while the mosaic files are compiled from the best quality strip files and distributed in 50 km × 50 km sub-tiles. The height in all the products is referenced to the WGS84 ellipsoid and in the map projection of Polar Stereographic North (EPSG:3413). Co-registration using IceSAT altimetry data is applied to mosaic files to improve absolute accuracy, but not to strip files, which only provide offsets in x, y, and z directions (termed as xyz offsets) in the metadata. The ArcticDEM is publicly available on the PGC website and Google Earth Engine ([earthengine.google.com](http://earthengine.google.com)). The ArcticDEM can capture the shape, extent, and expansion of an RTS as exemplified by Fig. 4. The ArcticDEM has been widely used in many studies such as measuring lava flows (Dai and Howat, 2017) and estimating river surface elevation (Dai et al., 2018).



Fig. 1. Ground photo of a retrogressive thaw slump in the Hornaday River Uplands, Northwest Territories, Canada (Photo: Trevor Lantz).

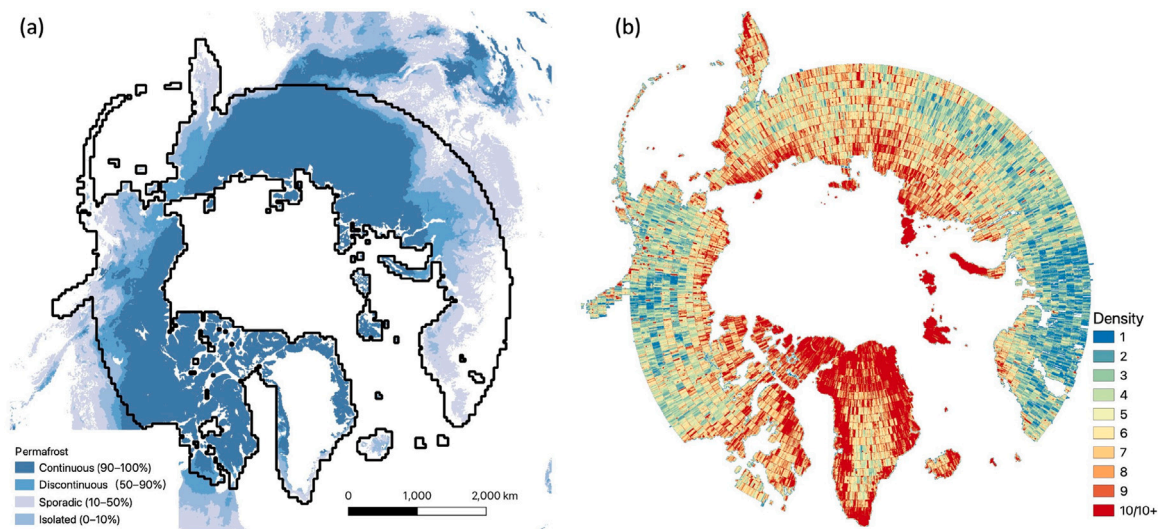


Fig. 2. The coverage and density of the ArcticDEM. (a) The black polygons outline the coverage of the ArcticDEM in the Arctic with permafrost extent (Obu et al., 2019) in the background. (b) The density of the ArcticDEM indicates the number of multiple temporal digital elevation models (DEM) available at each location.

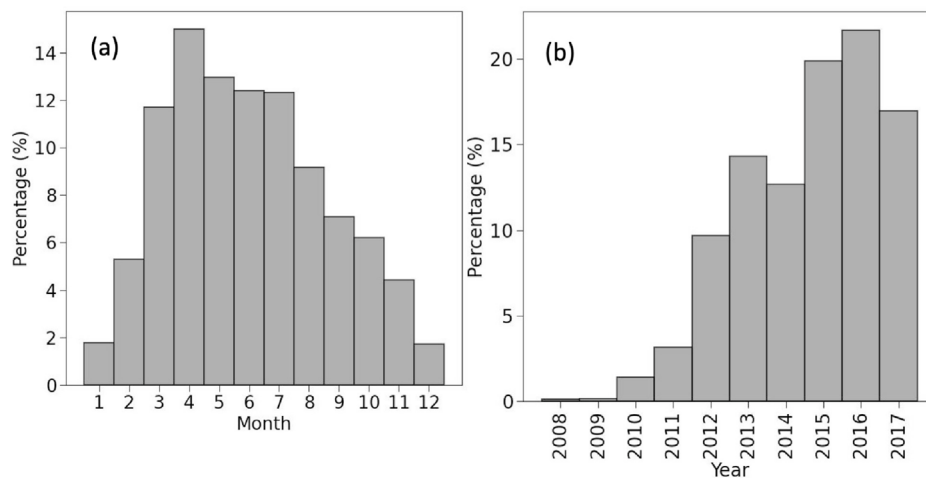


Fig. 3. Histograms of months (a) and years (b) of the ArcticDEM.

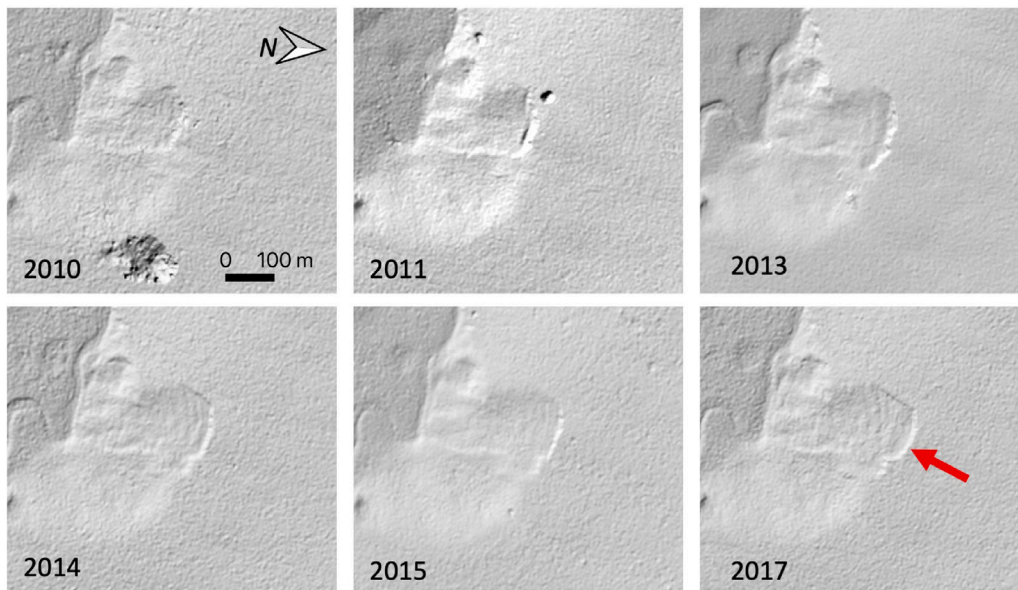


Fig. 4. Examples of a retrogressive thaw slump (RTS) in hillshades derived from DEMs in 2010, 2011, 2013, 2014, 2015, and 2017. The geometric center of this RTS is 68.882 °N, 150.959 °W. The red arrow indicates the RTS headwall, which moves upslope (North). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3. Methods

Our workflow included an automated pipeline to process the ArcticDEM data and perform feature extraction, a deep-learning-based object detection algorithm (i.e. YOLOv4), and implement a web-based crowdsourcing system (Fig. 5). The pipeline used two workstations and a supercomputer for downloading the ArcticDEM data and running feature extraction including elevation differencing, segmentation of elevation differences, extracting lines of narrow-steep slopes, identifying RTS headwall lines, and producing composite images (Section 3.1). The composite images were hillshades derived from the most recent DEM, with lines of narrow-steep slopes on them. These composite images were used as input images for locating RTSs using YOLOv4 (Section 3.2). We only kept YOLOv4 boxes (bounding boxes of objects identified by YOLOv4) at locations where RTS were present in all three features: polygons of elevation reductions, headwall lines, and YOLOv4 boxes. After filtering, we used a crowdsourcing system to validate/refine the filtered boxes and obtain bounding boxes of RTSs in the Arctic (Section 3.3). We chose regions in Alaska and the western Canadian Arctic and conducted some experiments to optimize the parameters of feature extractors and a YOLOv4 model.

#### 3.1. Automated pipeline for ArcticDEM processing and feature extraction

We developed an automated pipeline to process the ArcticDEM data and conduct feature extraction by utilizing two workstations in our lab and a supercomputer ([www.colorado.edu/rc](http://www.colorado.edu/rc)) on the campus of the University of Colorado Boulder (CUBoulder) (Fig. 5). The pipeline downloaded the ArcticDEM mosaic and strip files from the PGC website ([arcticdem.org](http://arcticdem.org)), unpacked them, and uploaded them to the supercomputer for feature extraction. On the workstations, xyz offsets, derived from co-registration based on IceSAT altimetry data, were applied to strip files before uploading. Due to the limitation of storage on our workstations (20 TB) and supercomputer (10 TB), the pipeline used a data management strategy to overcome this issue (detailed in Section 3.1.1). Communication between workstations and the supercomputer was through json files containing information such as a subset id and job statuses. The pipeline also monitored failed jobs on the supercomputer, then re-ran them on workstations. The heterogeneous nature of feature extraction in different grids caused varying demand

of CPU memory and computing time, which sometimes exceeded pre-allocated resources requested from the supercomputer and led to job failure.

##### 3.1.1. Data preparation and management

The pipeline divided the entire ArcticDEM domain into many grids (Fig. 6), grouped connected grids into subsets (Fig. 7), and took each subset as a whole to download and prepare data. The entire domain includes 58,667 grid cells, each of them is a basic unit for data processing and has a size of 20 km × 20 km, which well fitted the allocated resources of a computing job on the supercomputer. The pipeline grouped connected grids to subsets by rasterizing grids into a raster file, then using the region growing algorithm (Figs. 7 c and d). Each subset contains a few to 200 grids, depending on how many connected grids the algorithm could find. Because a strip or mosaic file overlaps several grids (Figs. 7 a and b), and each location is covered by multiple strip files (Fig. 2b), the pipeline treated downloading the ArcticDEM from the PGC website ([arcticdem.org](http://arcticdem.org)) for a subset of grids as one task. We did not directly use the ArcticDEM available on Google Earth Engine because the platform does not have the algorithms needed for feature extraction. Data preparation included downloading/unpacking strip and mosaic files, applying xyz offsets to strip files using a tool published by PGC (<https://github.com/PolarGeospatialCenter/pgcdemtools>), and uploading files to the supercomputer. The pipeline only downloaded and prepared mosaic and strip files that covered a few subsets of grids ahead of feature extraction due to the limitation of disk space on the workstations and the supercomputer. Once feature extraction for all the grids covered by a strip or mosaic file was complete, the corresponding file would be deleted on all machines to release disk space.

Before feature extraction, the pipeline cropped strip and mosaic files to the extent of each grid, applied masking data, and merged strip files. After cropping, the pipeline applied masking data including matchtag, DEM outlier, and surface water to strip files. Matchtag is a raster file produced by SETSM (Section 2) and indicates that a pixel is derived from a stereo match (1) or interpolation (0). We considered the DEM value of a pixel in a strip file as an outlier if it was greater than the corresponding DEM value in the mosaic file by 50 m or more. We masked pixels in the strip files if their matchtag was 0 or were considered as outliers. Because SETSM tended to produce inaccurate or incorrect DEM values over surface water, we utilized a dataset of

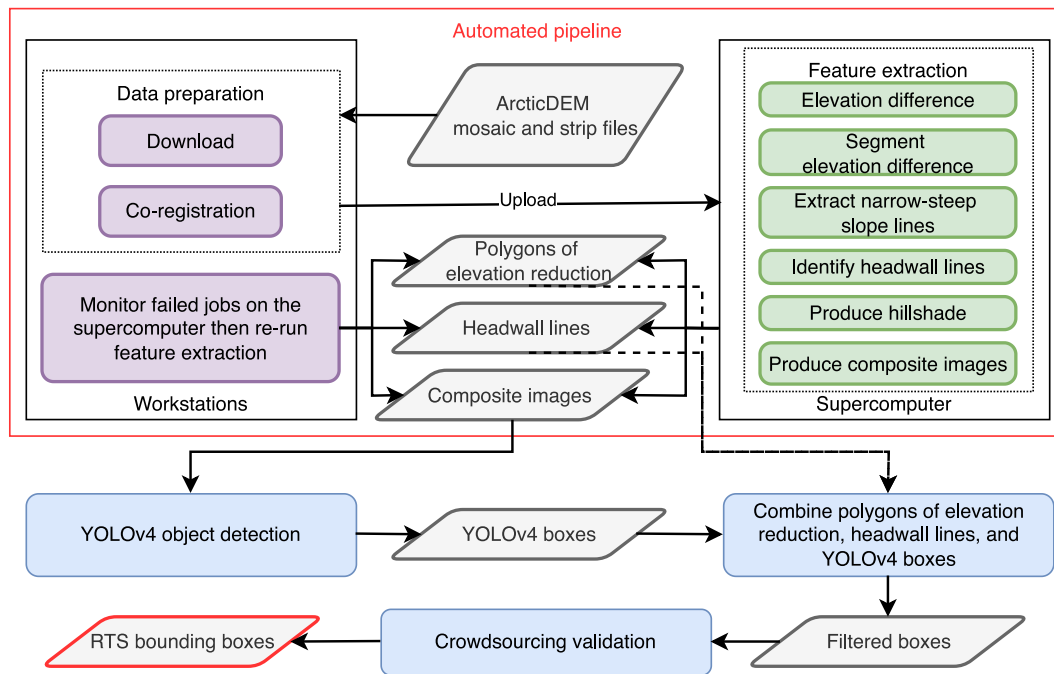


Fig. 5. The flowchart for identifying RTSs from the ArcticDEM using a combination of extracted features and deep learning (YOLOv4).

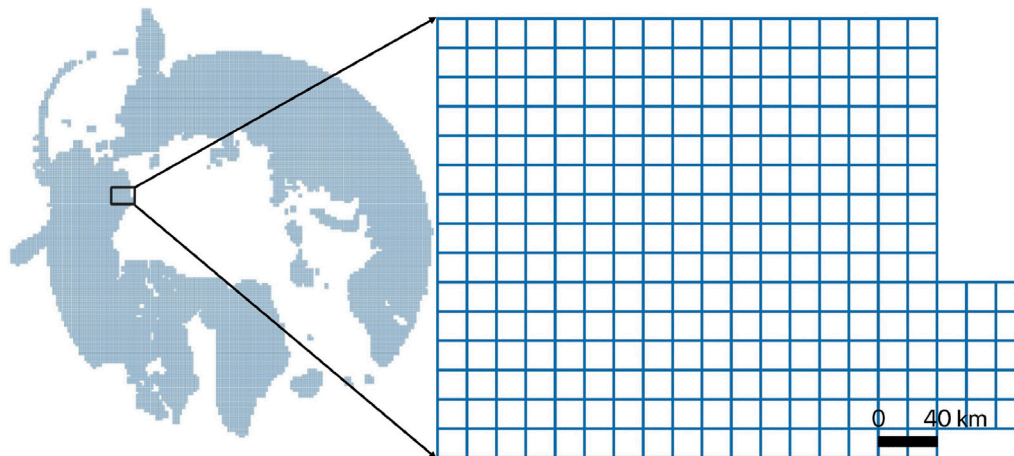


Fig. 6. Map of the study area showing grid system used for ArcticDEM processing. The entire ArcticDEM domain included 58,667 grid cells, each with a size of 20 km × 20 km.

global surface water (Pekel et al., 2016) to mask these pixels. After masking, strip files with the same scene ID and acquisition date were merged into one, and those acquired in the same year were also stacked by putting the layer with the acquisition date closer to July 1st on top. The pipeline used the Geospatial Data Abstraction Library (GDAL) and Python packages (Rasterio, GeoPandas, and Numpy) for reading/writing, cropping, masking, and merging.

### 3.1.2. Producing and segmenting elevation differences

The pipeline calculated the DEM difference at each pixel as:

$$D = DEM_n - DEM_o \tag{1}$$

where  $D$  is the elevation difference, and  $DEM_n$  and  $DEM_o$  are the most recent and oldest elevations for each pixel, respectively. To save disk space, rasters of elevation differences (e.g. Fig. 8a) were multiplied by 100 and saved to Int16, with a range from -327.68 m to 327.67 m. We assumed that elevation difference in the Arctic derived from the ArcticDEM would not exceed this range. We did not use DEM co-registration techniques based on the assumption of control surfaces (Nuth and Kääb,

2011; Dai and Howat, 2017) to co-register multi-temporal DEMs before differencing because the co-registration needs intensive computing and may eliminate real signals due to snow cover or DEM uncertainties.

In the next step, the pipeline segmented a raster of elevation difference and obtained polygons representing regions of elevation reductions. It segmented a raster using the quick shift algorithm (Vedaldi and Soatto, 2008), implemented in an open-source python package: scikit-image (Van der Walt et al., 2014). The raster was normalized to 8 bit (0–255) and tiled to many patches (around 1000 × 1000 pixels) to facilitate the segmentation, overcome the limitation of CPU memory, and leverage parallel computing resources. After segmentation, the raster was divided into many homogeneous regions represented by polygons (Fig. 8b). Some of these polygons had vertical or horizontal edges due to tiling. The steps to obtain polygons of elevation reductions included: (1) removing polygons whose area smaller than 120 m<sup>2</sup> (30 pixels) or where the average elevation difference within the polygons was greater than -2 m; (2) merging polygons that touch each other; (3) removing polygons that matched one of the following criteria: (a) larger than 1,000,000 m<sup>2</sup> (250,000 pixels), (b) the relative elevation

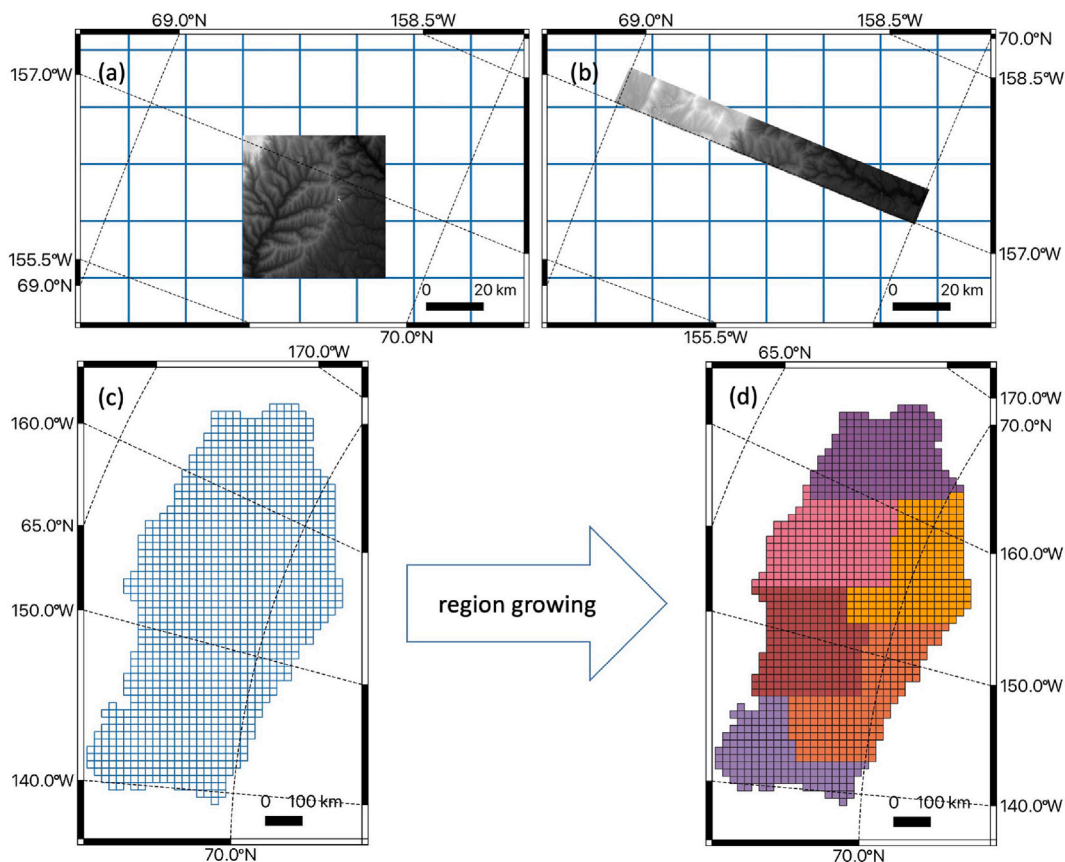


Fig. 7. Examples of ArcticDEM mosaic and strip files, and subsets containing connected grids. A mosaic (a) and strip (b) file covers several grids, respectively. (c) and (d) show that connected grids are grouped into subsets with different colors using the region growing algorithm.

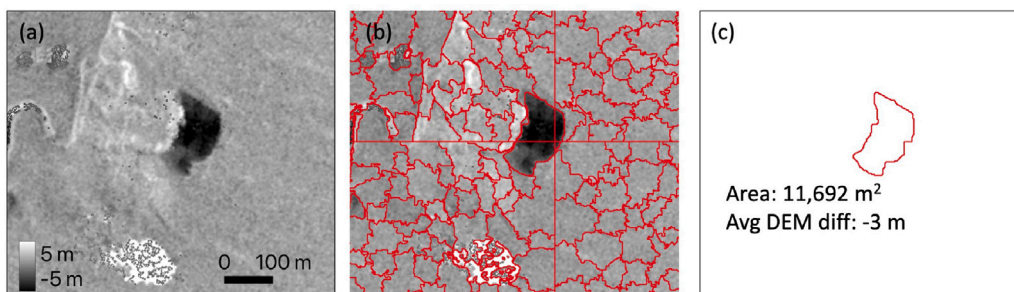


Fig. 8. An example of DEM differences and the corresponding segmented polygons. (a) is the result of DEM differencing, and red polygons in (b) are results after applying a segmentation algorithm to (a). (c) shows a polygon representing a region of elevation reduction after processing polygons in (b). These sub-figures have the same geographic extent and orientation as Fig. 4. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

differences are greater than  $-2$  m, (c) had a large and narrow shape, (d) contained more than 20 holes, (e) located on a slope steeper than  $20^\circ$ . In the step (2), we rasterized polygons into a binary raster then converted this layer to polygons, which was more efficient than checking if a polygon touched another polygon, and automatically removed the artifact (i.e. vertical or horizontal edges in Fig. 8b) caused by tiling. The relative elevation difference of a polygon was defined by

$$E_r = E_{in} - E_{su} \quad (2)$$

where  $E_{in}$  and  $E_{su}$  is the average of elevation differences within a polygon and its surroundings (a 20 m buffer zone), respectively. By using the relative elevation difference, we assumed that the regions surrounding an RTS are stable. This also minimized potential errors caused by inaccurate DEM values in the ArcticDEM. A polygon of an elevation reduction with a large area ( $>10,000$  m<sup>2</sup>) and narrow

shape (circularity  $< 0.1$ ) tends to result from changes in snow cover, vegetation, or positional errors of multi-temporal DEM data, instead of the development of an RTS. The circularity of a polygon is defined as:

$$\frac{4\pi S}{P^2} \quad (3)$$

where  $S$  and  $P$  is its area and perimeter, respectively. A polygon containing too many holes indicates that the DEM derived from stereopair images may contain errors and uncertainties. Since RTSs tend to occur on gentle slopes, we set a slope threshold to remove polygons that did not match this criterion, although the topography may already be disturbed by slumping. The average of slope within a polygon was calculated from the mosaic files of the ArcticDEM.

### 3.1.3. Delineating narrow-steep slopes

The pipeline delineated lines representing narrow and steep slopes from the strip files in different years. This was useful for identifying RTS

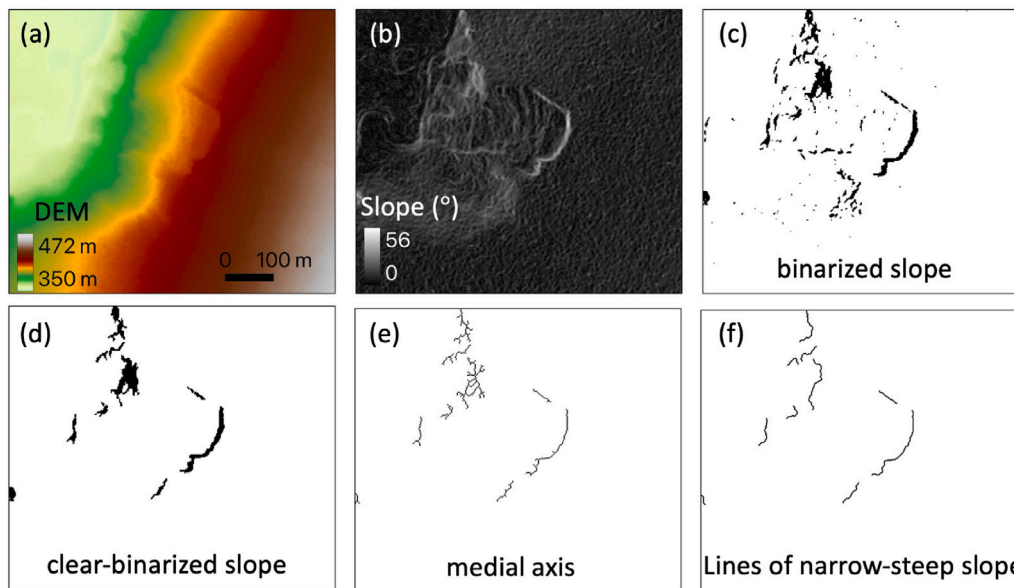


Fig. 9. Delineation of narrow-steep slopes from a DEM raster. (a)–(f) are a DEM raster, slope file, binarized slope, clear-binarized slope, medial axis, and final results of narrow-steep of slope, respectively. The former file is the input of the latter one. These sub-figures have the same geographic extent and orientation as Fig. 4.

headwalls which cause abrupt changes in elevation. The steps in this part of the workflow included: (1) converting a DEM raster (Fig. 9a) to a slope file (Fig. 9b) using GDAL, (2) binarizing the slope raster by assigning pixels with slope  $>20^\circ$  as 1 (Fig. 9c), (3) removing big ( $>50000 \text{ m}^2$ ) or very small ( $<200 \text{ m}^2$ ) regions (Fig. 9d), (4) extracting medial axes from the remaining regions (Fig. 9e), and (5) obtaining lines representing narrow and steep slopes (Fig. 9f). We did not apply matchtag to strip files before the delineation because it generated many invalid pixels similar to white noise that may have resulted in failure of the delineation process. In step (3), we used the measure module in a python package called scikit-image to group connected pixels into regions and obtain their areas, and remove any regions that were too big or small. We also used the morphology module in scikit-image to extract medial axes and obtain many line segments with one-pixel width. Subsequently, we converted medial axes to polygons and removed some of them if: (1) their length was greater than 4000 m or less than 30 m, (2) the maximum width of their medial axes was greater than 80 m, (3) or they contained more than 10 holes. In step (5), each pixel in the remaining medial axes would be considered as a node of a graph, and we used a python package NetworkX (Hagberg et al., 2008) to conduct graph analysis and obtain lines representing narrow-steep slopes using the Dijkstra's Method (Dijkstra et al., 1959), which is a commonly used method to identify the shortest weighted path in a graph from source to target locations.

### 3.1.4. Selecting headwall lines from narrow-steep slope lines

The pipeline used a statistical method by gradually expanding the buffer zone of a line and counting other lines in its surroundings (termed ripple statistics). The main idea is to check if a group of narrow-steep slope lines follows the pattern of an RTS expansion, that is, its headwall moves in one direction each year. We detailed the calculation as follows.

We assumed that a line of narrow-steep slope is  $L_c$  (e.g. the red line in Fig. 10), a collection of lines in its surroundings is  $L$ :  $\{L_1, L_2, \dots, L_j, \dots, L_m\}$  (e.g. the black lines in Fig. 10),  $L_c \notin L$ , a time array is  $T$ :  $\{Y_{L_c}\}$ , a collection of potential headwall lines is  $H$ :  $\{L_c\}$ , and a collection of the number of newly found lines within the buffer zone of  $L_c$  is  $S$ :  $\{s_1, s_2, \dots, s_j, \dots, s_n\}$  (Fig. 10), where  $Y_{L_c}$  is the acquisition year of the original DEM (merged from the strip files in the same year, see Section 3.1.1) for deriving  $L_c$ . Two attributes of  $L_j$ : its length ( $|L_j|$ ) and acquisition year ( $Y_{L_j}$ ), were be used in

the following calculation. We gradually increased  $i$  from 1 to  $n$ , for each  $i$ , we calculated  $s_i$  (initialized as 0) by following steps: (1) obtain  $B_i = L_c.buffer(i \times \delta)$ ; (2) for each  $L_j \in L$ , if it is within  $B_i$ , then add it to a collection  $M$ ; (3) if  $M$  is empty, then increase  $i$  and go back to (1), otherwise, for each  $L_k \in M$ , if  $|L_c| \times 0.3 \leq |L_k| \leq |L_c| \times 3$  and  $Y_{L_k} \notin T$ , add 1 to  $s_i$ , append  $Y_{L_k}$  to  $T$  as the last element, and append  $L_k$  to  $H$  as the last element; (4) for each  $L_k \in M$ , remove it from  $L$ , then empty  $M$ . We set  $n = 50$ ,  $\delta = 2 \text{ m}$ , and used a python package shapely (shapely.readthedocs.io) for buffering and checking if a line is within  $B_i$ .

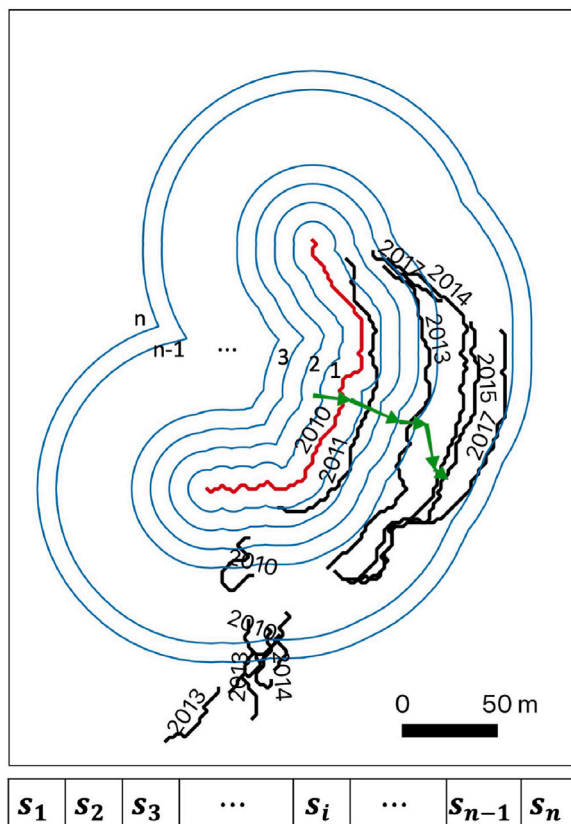
The pipeline decided if  $L_c$  is a headwall line based on variables derived from  $S$ ,  $T$ , and  $H$ . We assumed that  $O$ :  $\{O_1, O_2, \dots, O_d\}$  is the collection of geometric centers of lines in  $H$ ,  $d$  is the total number of lines in  $H$ , and  $O_1$  to  $O_d$  indicates how the headwall of an RTS moves (the green arrows in Fig. 10). We calculated sinuosity of  $O$  by:

$$Sin = \frac{|O_1O_2| + |O_2O_3| + \dots + |O_{d-1}O_d|}{|O_1O_d|} \quad (4)$$

where  $|O_aO_b|$  is the length of the straight line connecting two geometric centers  $O_a$  and  $O_b$ , and  $a, b \in (1, d)$ . We assumed that  $A$ :  $\{A_1, A_2, \dots, A_z, \dots, A_{d-1}\}$  is the collection of angles between  $O_{a-1}O_a$  and  $O_aO_{a+1}$ , where  $A_z \in [0^\circ, 180^\circ)$ . We assumed  $r$  is the number of non-zero values in  $S$ . We considered  $L_c$  a headwall line if all the criteria were satisfied: (a)  $r > 1$ ; (b) the years in  $T$  is monotonic increasing or decreasing; and (c)  $Sin < 2$  and the minimum value of  $A$  is greater than  $90^\circ$ .

### 3.1.5. Producing composite imagery

The pipeline produced composite imagery (e.g. Figs. 11a and b) by rasterizing lines of narrow-steep slopes (Section 3.1.3) with hillshades derived from the most recent strip files. We did not apply the mask of matchtag and global surface water to the strip files before converting DEM to hillshades because we want to keep all information from the ArcticDEM. The width of each line is one-pixel, and the lines derived from later years would overwrite those from early years if they were at the same locations. We assigned different colors for the lines in different year as shown in Fig. 11c. We used these imagery as the input for deep learning algorithms because we anticipate that locating RTSs from composite imagery is more accurate and easier than from hillshades only.



**Fig. 10.** Illustration of the identification of a headwall line from lines of narrow-steep slopes. The red and black lines are lines of narrow-steep slopes (Section 3.1.3), and the red one is the line being checked if it is a headwall line. The blue curves are boundaries of buffer polygons at different steps (1 to  $n$ ). The green arrows indicate the movement of headwall lines from 2010 to 2017.  $s_1$  to  $s_n$  are counts of newly contained lines by the buffer polygons at different steps. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3.2. Identifying RTSs using YOLOv4 and extracted features

We used an object detection algorithm to locate RTSs on the composite imagery in the ArcticDEM domain and obtain bounding boxes of RTSs. Object detection is a category of computer vision algorithms for locating objects in images or video and obtaining corresponding bounding boxes and classes. The object detection algorithm we used was YOLOv4 (Bochkovskiy et al., 2020), which is an improved version of YOLO (Redmon et al., 2016). YOLO means “You Only Look Once” and is a super-efficient algorithm, comparing with other state-of-the-art object detectors (Bochkovskiy et al., 2020), that allowed us to finish the processing of a huge dataset for the entire region with limited computing resources in a few days.

We trained a YOLOv4 model using training polygons drawn in QGIS (www.qgis.org) and composite imagery. We used RTS locations (points) published by Bernhard et al. (2022a) as guidance then delineated 563 polygons (e.g. the dashed polygon in Fig. 11b) of RTSs on these imagery and considered them as positive training polygons. These RTS points were detected from TanDEM-X-derived DEM using a change detection algorithm (Bernhard et al., 2020). By following the practice in Huang et al. (2020), we also drew some negative training polygons to reduce false positives when we originally focused on Alaska and fine-tuned the YOLOv4 model. The positive polygons represent RTS headwalls and parts of the slump scar, as we cannot distinguish slump scars using hillshades. The number of positive training polygons is less than that of points published by Bernhard et al. (2022a) because of the discrepancy between TanDEM-X and ArcticDEM

observation period, which varies on location (Figs. 2b and 3). We developed an automated routine that can convert training polygons and the composite imagery into datasets for training YOLOv4. We used a pre-trained model ([https://github.com/AlexeyAB/darknet/releases/download/darknet\\_yolo\\_v3\\_optimal/yolov4.conv.137](https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.conv.137)) to initialize the YOLOv4 model and trained it with the default settings in Bochkovskiy et al. (2020), except that we set the number of classes to 1 because we only had one class (i.e. RTS). We randomly divided the training data into training (90%) and validation (10%) sets. The training loss stabilized at around 0.34 after 4000 iterations, and mean Average Precision (mAP) of the validation set was 85%, indicating a good training performance. Due to the limitation of computing resources and the large scale of data, we did not conduct various experiments to optimize the YOLOv4 model for the entire ArcticDEM domain.

We used the trained model to locate rapid expanding RTSs in the entire ArcticDEM domain from the composite imagery and obtained bounding boxes and the corresponding confidence. The composite imagery was split into many patches, each with size of  $320 \times 320$  pixels and an overlap of 80 pixels with adjacent ones. We set a threshold of 0.5 for non-maximum suppression, meaning that multiple bounding boxes have 50% overlap with each other would be removed and only that with the maximum confidence would be kept.

In the final step of this process, we combined the bounding boxes output by YOLOv4 and extracted features including polygons of elevation reductions and headwall lines to remove potential false positives as many as possible. We selected headwall lines if polygons of elevation reductions touched or intersected their 20-m buffer zones. We then compared bounding boxes with selected headwall lines and kept those bounding boxes if the lines touched or intersected their 10-m buffer zones. After this, we also manually checked the composite imagery outlined by the bounding boxes and removed those that did not have RTS features within the boxes.

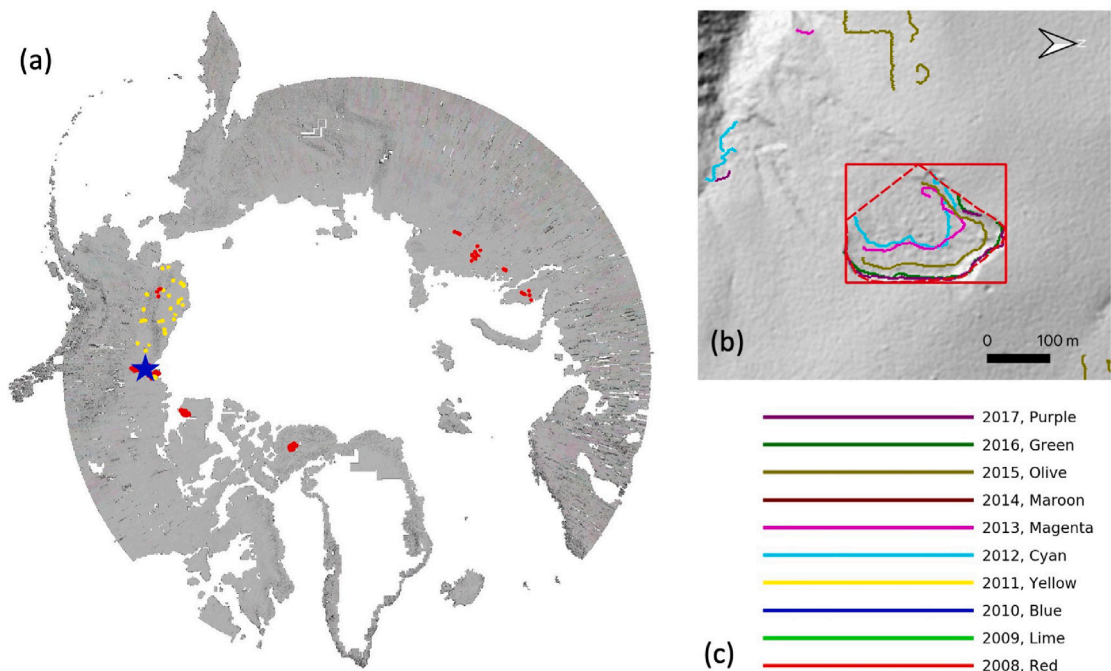
### 3.3. Validating and refining RTSs using crowdsourcing

We developed an online crowdsourcing system that integrates a user input form, composite imagery, and interactive maps of high-resolution imagery to validate and refine RTS bounding boxes (Fig. 12). The system allowed users around the world to validate, and improve the accuracy, of mapping results obtained using our method (Section 3.2). After logging in to this system, contributors can use panel (c) as a main window to check the results and DEM imagery, panel (a)&(b) for additional high-resolution satellite imagery, and panel (d) to indicate if the feature was correctly identified as an RTS, by selecting one of the following options: Yes, No, High Confidence, Medium Confidence, and Low Confidence.

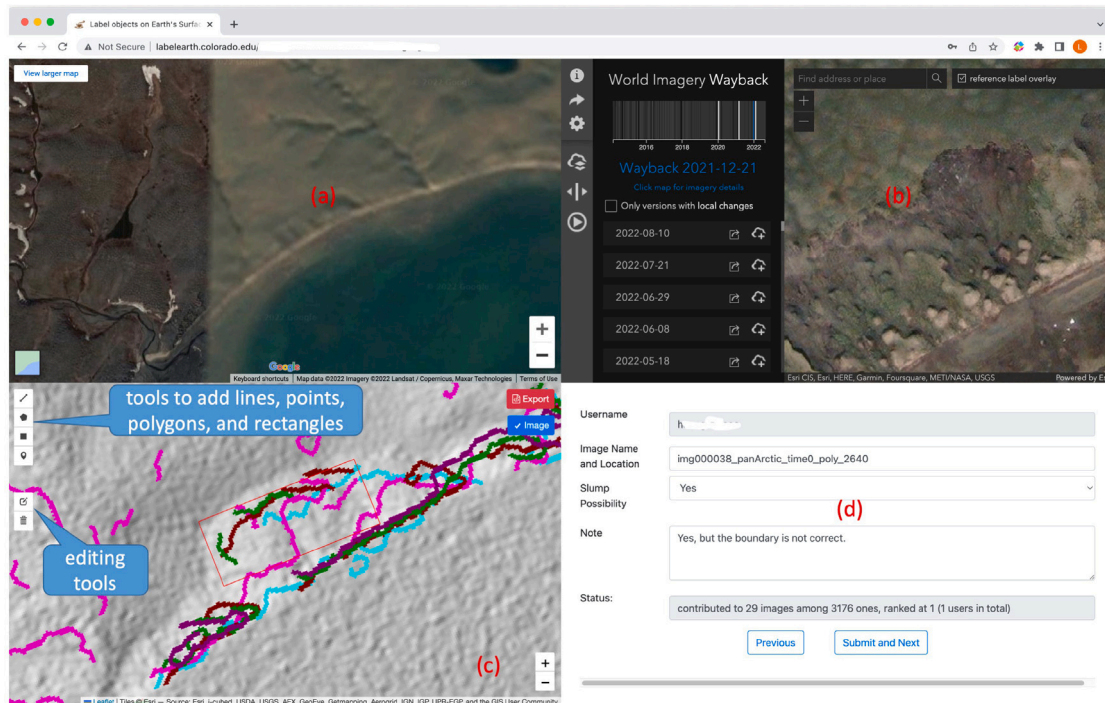
We used Javascript to develop the frontend, and Django (www.djangoproject.com, version 4.0.3) for the backend, with PostgreSQL (www.postgresql.org, version 14.5) as the database. We used the Leaflet (leafletjs.com, version 1.8.0) for displaying composite imagery, bounding boxes, and tools for drawing as well as editing. Each bounding box would be validated up to three times by different users. We also encouraged people to add bounding boxes for false negatives shown in panel (c). The crowdsourcing system is hosted in the server at the Cooperative Institute for Research in Environmental Sciences, University of Colorado Boulder and can be accessed by visiting <http://labelarth.colorado.edu>. We shared the crowdsourcing systems through different channels to encourage participation including department email lists at CUBoulder, Cryolist, Twitter, and the International Permafrost Association (IPA) RTS InTrain Action Group. Instructions on the use of the crowdsourcing system and contributions are at <https://yghlc.github.io/validate-thaw-slump/>.

We defined a strategy to obtain results that only contain true positives and the bounding boxes or polygons newly added by volunteers after crowdsourcing validation. We assigned values: 1.0, 0.75, 0.5, 0.25, and 0 for the mapper confidence: Yes, High, Medium, Low,





**Fig. 11.** Training polygons and composite imagery. (a) shows the overview of composite imagery in the ArcticDEM domain and the distribution of training polygons, red and yellow polygons are positive and negative training data, respectively. (b) shows an example of composite imagery, the positive training polygon and the bounding box of an RTS whose location is marked by a blue star in (a). The time period associated with each narrow-steep line is shown in (c). The geometric center of this RTS is 68.138 °N, 135.551 °W. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



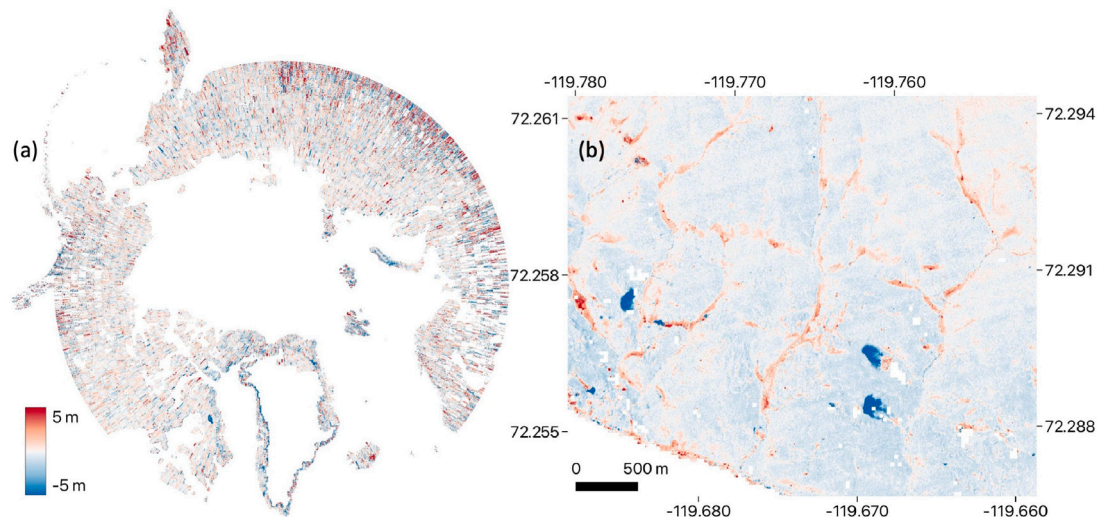
**Fig. 12.** The main interface of the crowdsourcing system shows imagery from Google satellite imagery (a), ESRI World Imagery Wayback (b), and the composite imagery as well as the bounding box of a retrogressive thaw slump. Panel (d) is the input form. Tools in Panel (c) allow users to edit a bounding box if it is not accurate or add a bounding box for a missed RTS.

and No, respectively. For a bounding box generated by YOLOv4, we calculated its average mapper confidence input from volunteers and considered it as a true positive if the average mapper confidence greater

than or equal to 0.5. We used non-maximum suppression to make sure only one box or polygon is at the location of an RTS if multiple volunteers repeatedly added polygons for an RTS missed by YOLOv4.

**Table 1**  
List of features extracted from ArcticDEM.

Feature	Size (GB)	Count	Type	Description
elevation differences	5,888	–	raster	Pixel-wise elevation differences obtained by comparing the most recent DEM and oldest one
polygons of elevation reductions	251.4	104,125,249	vector	regions of elevation reductions
lines of narrow-steep slopes	1,928	1,929,188,730	vector	lines representing narrow and steep slopes
potential RTS headwall lines	38.4	44,048,963	vector	lines selected by ripple statistics from narrow-steep slope lines
composite imagery	5,574	–	raster	rasterized lines of narrow-steep slopes on hillshade derived from the most recent DEM.



**Fig. 13.** The overview (a) of pixel-wise elevation differences in the Arctic, not including the Greenland Ice Sheet, and an enlarged area showing several RTS (b) in Banks Island, Canada.

During non-maximum suppression, we assigned higher scores to the added or modified polygons, which means any YOLOv4 boxes would be deleted if they overlap user-input polygons.

## 4. Results

### 4.1. Features extracted from ArcticDEM

Features derived from the ArcticDEM include pixel-wise elevation differences, polygons of elevation reductions, lines of narrow-steep slopes, potential RTS headwall lines, and composite imagery (Table 1). Pixel-wise elevation differences (e.g. Figs. 8a and 13) and composite imagery (e.g. Fig. 11b) are rasters in the GeoTIFF format with a spatial resolution of 2 m, the same as strip files. Polygons of elevation reductions (e.g. Fig. 8c) represent regions of elevation reductions with size in a defined range (see Section 3.1.2) and are in the format of GeoPackage. Lines of narrow-steep slopes (e.g. Fig. 9f) indicate locations where slopes are greater than 20°, such as cliffs, river banks, and RTS headwall. Those with a temporal pattern similar to the movement of RTS headwall (e.g. Fig. 10) were selected by using ripple statistics and considered as the lines of potential RTS headwall (Section 3.1.4). These features were extracted from the ArcticDEM for identifying RTSs but some of them, especially elevation differences, can be used for other research. As shown in Fig. 13, elevation differences in most regions in the Arctic are near zero, which is reasonable as most regions were not experiencing changes, while the spots with significant elevation changes are localized and need further investigation.

### 4.2. The bounding boxes output by YOLOv4

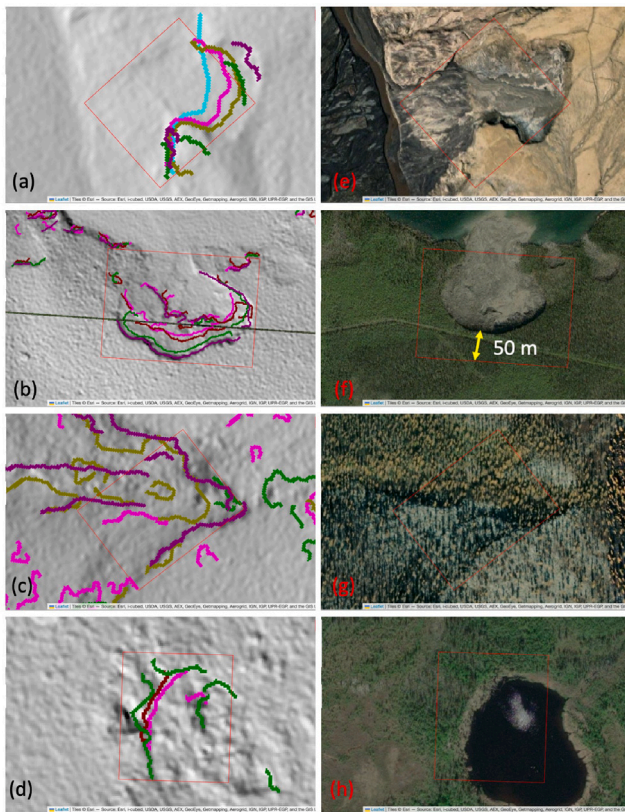
A total of 513,295 bounding boxes were directly output by YOLOv4, and 15,290 ones remained after combining polygons of elevation reductions and potential headwall lines (Table 2). After manual inspection,

**Table 2**  
The count of bounding boxes in different steps.

	YOLOv4	Combining extracted features	Manual selection
count	513,295	15,290	3,176

3176 bounding boxes were kept as inputs for the crowdsourcing system. Comparing the 563 polygons of RTSs in training data (Section 3.2) with the 513,295 bounding boxes, 527 (93.6%) correctly identified and 36 (6.4%) of them were missed by the trained YOLOv4, indicating a good training accuracy but no guarantee of a good transferability. Filtering by extracted features and manual inspection only kept 123 (23.3%) of the 527, suggesting a discrepancy between extracted features and the features automatically learned during the training of YOLOv4. The extracted features successfully reduce the number of bounding boxes to a manageable level (i.e. 15,290) but have a cost of removing some true positives whose characteristics do not match these features. This is a trade-off between feasibility and the comprehensiveness of an RTS inventory in the entire ArcticDEM domain and requires an improvement of extracted features in the future (Section 5.5).

The trained YOLOv4 identified some RTSs (true positives) and other land covers (false positives) as exemplified by Fig. 14. The bounding boxes in Fig. 14 tilt due to the difference of map projections between data processing (EPSG:3413) and the crowdsourcing system (EPSG:3857). RTSs can be easily observed from composite imagery (Figs. 14 a and b) and satellite imagery (Figs. 14 e and f). The composite imagery also shows the movement of RTS headwall. For example, Fig. 14a shows the positions of the entire or parts of headwall in 2012 (Cyan), 2013 (Magenta), 2015 (Olive), 2016 (Green), and 2017 (Purple). False positives can be caused by many land cover types such as forest (Figs. 14 c and g) and surface water (Figs. 14 d and h).



**Fig. 14.** Examples of bounding boxes (red polygons) output by YOLOv4 and the corresponding composite imagery and satellite imagery shown in online maps. (a)–(d) are the composite imagery, and (e)–(h) are the corresponding satellite imagery. These are screenshots from the crowdsourcing system and do not have scale bars. The sizes of bounding boxes from (a) to (d) are 122 m × 136 m, 250 m × 350 m, 122 m × 162 m, and 128 m × 148 m, respectively. Their geographic centers are (79.440 °N, 86.790 °W), (67.520 °N, 131.102 °W), (66.274 °N, 97.293 °E), and (67.279 °N, 133.125 °W), respectively. The bi-directional arrow in (f) indicates the difference of headwall position between the composite and satellite imagery due to the differences in their acquisition times. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Table 3**  
Statistics of validation times to the bounding boxes.

Validation times	0	1	2	3
count of bounding boxes	0	3070	65	41

**4.3. Responses to the crowdsourcing system**

Two months after setting up the crowdsourcing system and advertisement, all the bounding boxes (Table 2) had been validated at least once by the authors team and volunteers (Table 3). The author team contributed most (>90%) of the validation. A total of 38 volunteers logged in to the crowdsourcing system, and 24 of them validated the bounding boxes. The system also received comments for bounding boxes from the volunteers, and some comments describe the cause of the false positives. For instance, (1) “snow drift in themokarst gully?”, (2) “icesheet”, (3) “just a steep Bluff”, (4) “deep valley, bad imagery”, (5) “Gully, might be an old inactive RTS”, (6) “dendritic watershed systems”, and (7) “poor satellite imagery quality but most likely not an RTS”. These comments provide background knowledge and are helpful for the improvement of future mapping efforts.

Among the 3176 bounding boxes (Section 4.2), 647, 387, 991 of them have mapper confidence greater than or equal to 0.75, 0.5, 0.25, respectively (Table 4). 35.3% of the bounding boxes did not locate RTSs but some other land cover or landforms. Those bounding boxes

**Table 4**  
The statistics of the RTS mapper confidence from the crowdsourcing system.

Avg mapper confidence	0	(0, 0.25)	[0.25, 0.5)	[0.5, 0.75)	[0.75, 1]	Total
count	1123	28	991	387	647	3176
percent	35.3%	0.9%	31.2%	12.2%	20.4%	100.0%

with low mapper confidence (<0.5) need additional imagery or ground knowledge for validation. A total of 2248 polygons were added to the system when volunteers found RTSs missed by YOLOv4 or modified inaccurate bounding boxes; 771 of the 2248 polygons overlap the original bounding boxes because (1) the system created a new polygon when volunteers modified the original box and (2) volunteers may repeatedly add polygons for missed RTSs because the system only shows the composite imagery and one bounding box in the center.

**4.4. Identified RTSs after crowdsourcing**

A total of 2494 RTSs were identified from the ArcticDEM across the Arctic after crowdsourcing validation, among them, 305 and 367 ones with mapper confidence greater than or equal to 0.75 and 0.5, respectively, after the non-maximum suppression. The remaining 1822 were added or modified from the original YOLOv4 boxes by volunteers during validation. Most RTSs are in the continuous permafrost zone and a few of them are in the discontinuous zone (Fig. 15a). These RTSs spread out across the Arctic but tend to cluster in some locations, as shown in the 100 km × 100 km grid map (Fig. 15b). There is also a cluster of RTSs in Northeast Greenland (Fig. 17), where permafrost and its thaw has not been adequately investigated and only a few RTSs were reported (Cable et al., 2018; Pastor et al., 2021), although the size and number of RTS in this cluster are not comparable to those in Canada and Siberia. The width and length ( $length \geq width$ ) of RTS bounding boxes ranges from 16.3 m to 885.8 m and 19.2 m to 1317.9 m, with mean values of 133 m and 181.5 m, respectively (Fig. 16), indicating the advantage of very high resolution data for identifying relatively small RTSs.

**5. Discussion**

**5.1. RTSs identified from ArcticDEM**

Our workflow using the ArcticDEM identified many RTSs in the Arctic and provided an important dataset for future studies. To the best of our knowledge, this is the first time that very high resolution (<5 m) remote sensing data has been used to map RTSs at a pan-Arctic scale. There are several lessons learned from this process that are relevant to similar studies or research that involves processing of large earth observation datasets. The bounding boxes of RTSs enrich the dataset of known RTSs in the Arctic and can be part of the training data for future mapping and monitoring of RTSs because the first problem for monitoring RTSs and assessing their environmental impacts at the pan-Arctic scale is the locations of all RTSs. Our results also reveal several clusters of RTSs, such as the one in Greenland (Section 4.4). The locations of those clusters in Canadian and Russian Arctic match previous studies (Lewkowicz and Way, 2019; Huang et al., 2022; Bernhard et al., 2022b). The mapper confidence (Section 4.3) for each RTS represent uncertainties associated with remote sensing data used during identification and validation. Researchers should be cautious when they include those RTSs with low mapper confidence into their studies.

Our results do not represent a comprehensive inventory of RTSs in the Arctic because many RTSs were missed due to the limitations related to the ArcticDEM and our method (see Section 5.5). For example, in sub-regions of Alaska where Nitze et al. (2018) and Swanson (2021) identified a few hundred RTSs, our workflow only identified 23 RTSs

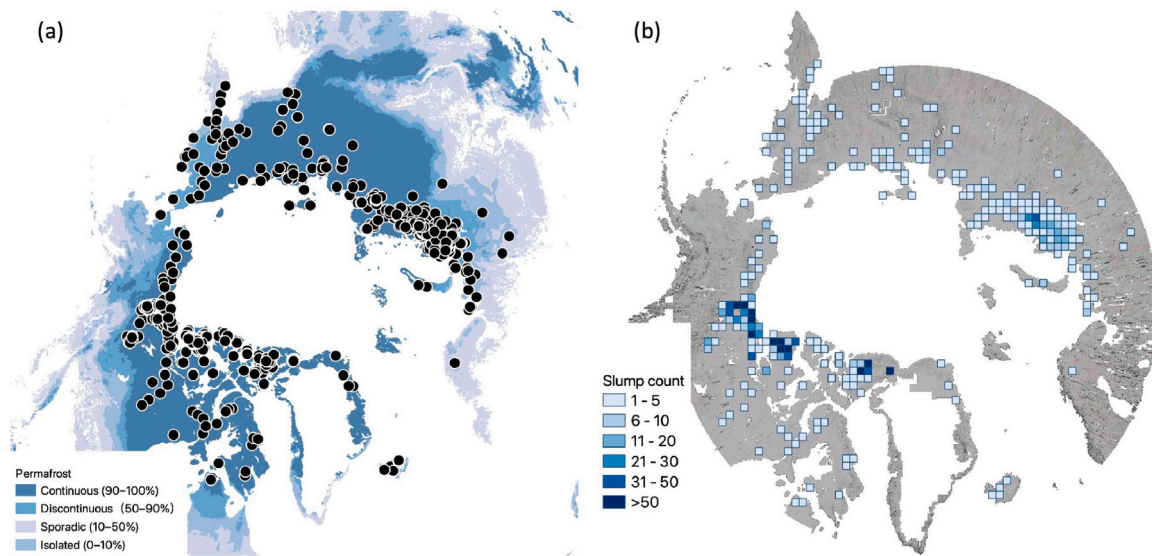


Fig. 15. The distribution (a) of the RTSs identified in the Arctic and their number (b) in 100 km × 100 km grids.

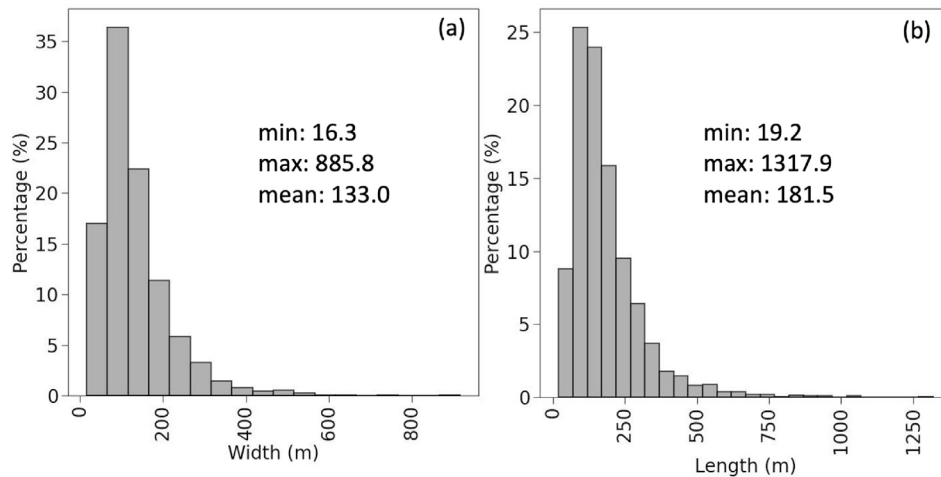


Fig. 16. The histogram and statistics (minimum, maximum, and mean values) of the width (a) and length (b) of the bounding boxes of identified RTSs.

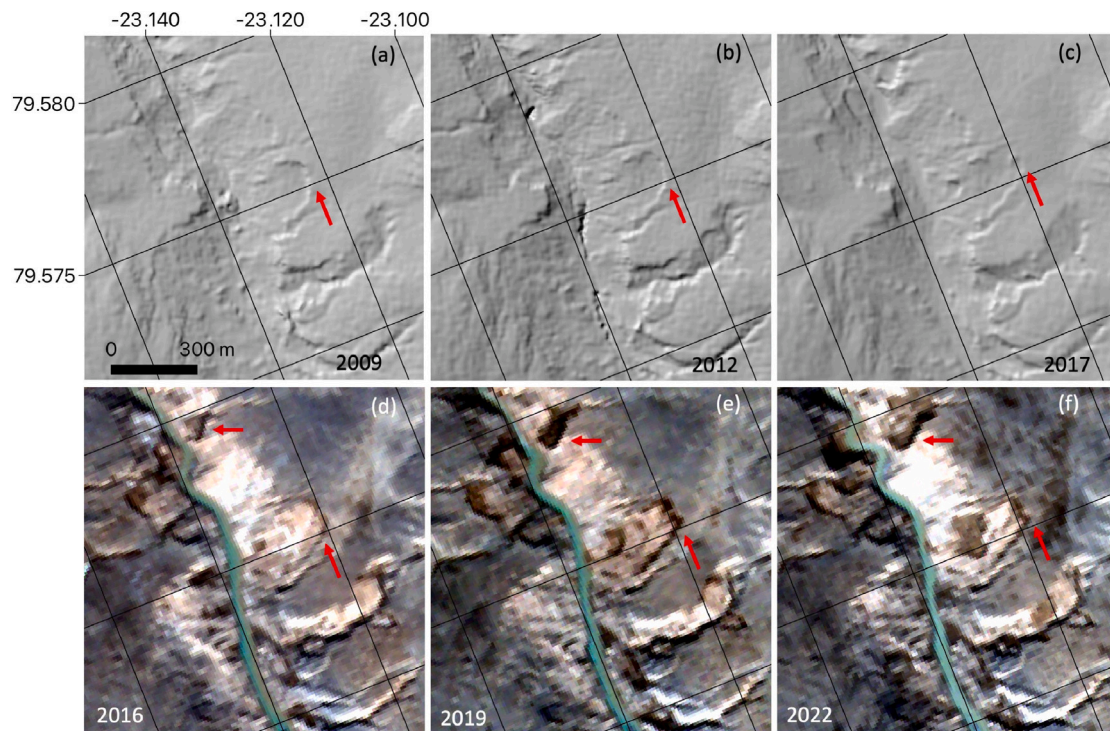
in Alaska. Runge et al. (2022) applied an automated detection method to satellite imagery in North Siberia and identified 50,895 potential RTSs, which is well beyond the number of RTSs found in this study. Because of missed RTSs in the results, we did not conduct analysis of the spatial distribution or controlling factors, to avoid misleading conclusions. RTSs are small and dynamic features and compiling an complete inventory remains a significant challenge that requires effort from permafrost and remote sensing communities. Despite the fact that our workflow only identified a fraction of the RTS in the study domain, our experience offers several insights relevant to future efforts to map thermokarst features across the circumpolar north.

### 5.2. The advantages of ArcticDEM and the challenges to processing it

The ArcticDEM is a high resolution, multi-temporal, and open-access DEM covering the Arctic and is a fundamental dataset for many research in the Arctic. The spatial resolution of 2 m provides details on the topography such as buildings, landslides, and microtopography. Multi-temporal observations using the ArcticDEM enable the detection of changes on topography over time. The open-access policy facilitates many studies that might not happen if payment or permission was required for accessing the ArcticDEM.

The extracted features (Section 4.1) derived from the ArcticDEM in this study also can be used in other research or future work to identify permafrost landscape change. The pixel-wise elevation differences with a resolution of 2 m can be used to explore surface elevation change. Lines of narrow-steep slopes represent the locations of abrupt slope changes on topography such as river banks and cliffs. Composite imagery (Section 3.1.5) can provide useful or supplement information on topography when satellite imagery is not available at certain locations and times.

Since the ArcticDEM is derived from optical imagery, it presents several limitations and challenges when using it to investigate details on topography. Optical sensors can be affected by atmospheric conditions such as cloud and haze, and cannot penetrate vegetation canopy that covers the Earth’s surface. Therefore, the ArcticDEM contains regions with invalid data and does not represent the height of bare-Earth, but that of anything on the Earth’s surface. Changes in vegetation and snow over time also create noise and uncertainties, which complicate detecting changes in topography. The algorithm (i.e. SETSM) for producing ArcticDEM also has problems over surface water and tends to produce invalid or incorrect DEM values, also causing challenges. The size of ArcticDEM is also a challenge because most researchers do not have the IT facility to host and process it.



**Fig. 17.** RTS examples in Northeast Greenland. (a)–(c) are hillshades derived from ArcticDEM of 2009, 2012, and 2017, respectively, and show the expansion of an RTS headwall (pointed by red arrows), which passed the latitude-longitude grid line (black). (d)–(f) are Sentinel-2 images from 2016 to 2022 and show the expansion of a few RTSs (red arrow) in the upper region of (d)–(f) emerged after 2012 and is not shown in (a) and (b). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 5.3. The necessity of combining deep learning and extracted features

Combining deep learning and extracted features is required when limited training data is available for large regions. The main advantage of deep learning is the automated feature engineering (LeCun et al., 2015), allowing people to handle computer vision tasks without skills and domain knowledge for designing feature extractors. Sometimes, deep learning can automatically learn better features from data than human classifiers, but may have a problem of overfitting or low generalization. In this study, the trained YOLOv4 identified 524,240 bounding boxes (Section 4.2), which is beyond a manageable level for manual inspection and crowdsourcing validation. The huge number of bounding boxes indicates a potential problem of generalization in the trained model. Two months after the crowdsourcing system was set up, responses from volunteers or the permafrost community were very limited (see Section 4.3). Therefore, manual validation of the 524,240 bounding boxes is infeasible. Extracted features including polygons of elevation reductions and RTS headwall lines can also be used to identify RTS, but these approaches also result too many features to feasibly validate. For example, if combining polygons of elevation reductions and RTS headwall lines using the approach in Section 3.2, we obtained 8,316,110 lines, far beyond a manageable level.

### 5.4. Addressing the computational challenges for large scale of ArcticDEM

Many technical problems caused by limited computing resources and heterogeneity in the large scale data were encountered when extending our algorithms from local regions to the entire ArcticDEM domain. The first technical challenge was the lack of disk space to store the ArcticDEM (~200 TB) and intermediate data as well as final results. We had access to a supercomputer for this project, but the quota of storage was limited. We tried three solutions including (1) trying to use the supercomputer producing the ArcticDEM, but we did not obtain the corresponding permission; (2) using Google Earth

Engine, but we found some algorithms we needed were not available on that platform; (3) developing a pipeline that only downloaded a portion of the ArcticDEM to CUBoulder, then immediately removed it after processing. Re-locating the 200 TB ArcticDEM from PGC to our workstations was time-consuming but feasible, thanks to the high-speed internet on campus. The second challenge was caused by the heterogeneity of ArcticDEM in the Arctic, which resulted in many failed jobs and instability of the pipeline. Different grids required different computing time and CPU memory, which often exceeded pre-allocated resources requested from the supercomputer and lead to job failure. To reduce the queue time of a computing job, we tried to request the minimum resources required for each job. For the failed jobs on supercomputer, we set up a monitoring program to identify them and then re-run them on the workstations, that had enough CPU memory and could allow a job to run for a long time. At the early stage of running the pipeline, it stopped working quite often because of unexpected situations in new grids. We had to investigate the problems and added python scripts to overcome the new problems. In the later stages of data processing the pipeline was quite stable.

Some technical challenges related to efficiency, crucial for processing big data, were also resolved during the processing and development of the pipeline. For example, (1) Using “gdal\_polygonize.py” (available in GDAL) to convert segmented results from rasters to polygons (Section 3.1.2), took more than 10 h or even a few days if there are a lot of segmented regions with complex shapes. Luckily, in the new version (3.4.2) of GDAL released in 2022, the algorithm for polygonization has been optimized. As a result, upgrading GDAL to the newest version likely will significantly improve the efficiency of producing polygons of elevation reductions. (2) Writing a vector file to disk using GeoPandas is time-consuming if the file is large; so we kept vector data in CPU memory as much as possible and avoid I/O operations. (3) The step to find adjacent polygons (Section 3.1.2) is also time-consuming, and we adopted a method to “rasterize polygons then polygonize it” to automatically merge polygons connected each other. (4) Obtaining the

areas of regions after segmentation using polygonizing also is time-consuming; we used the “regionprops” function in the measure model of scikit-image to get properties of segmented regions, which is much more efficient. To fully utilize computing resources, we implemented all the algorithms for feature extraction in parallel.

### 5.5. Limitation and future work

The main limitation of this study is that the final result misses many RTSs and contains false positives. We attribute these errors to four main causes. (1) For some locations or regions, the ArcticDEM does not have enough observations to capture the retreating behavior of an RTS; (2) the parameters for extracting features were chosen during experiments in western Canadian regions and may not fit to the characteristics of all RTSs in the entire ArcticDEM domain, especially those in the Russian Arctic; (3) YOLOv4 was trained with limited training data, which likely resulted in some false negatives and false positives; (4) available computing resources were not sufficient to conduct various experiments with different combinations of parameters and deep learning architectures. The method only identified rapidly expanding RTSs during observation periods and tended to ignore those relatively stable or expanded a little bit. Improvements in the future should focus on building a more accurate and efficient deep learning model to locate RTSs by adding more training data or choosing a better architecture. Feature extraction is not needed if a deep learning model can achieve a high mapping accuracy. Alternatively, future work can focus on select regions across the Arctic then fine-tune the parameters for manual-feature engineering (feature extraction) and find the best parameters for representing characteristics of all potential RTSs.

The percentage of false positives is still high after combining extracted features and YOLOv4 outputs. Despite the fact that we used a global surface water dataset as mask, many false positives occurred over surface water. This suggests that future inventories could be improved by using a higher resolution and more accurate surface water product. Hopefully, a new version of ArcticDEM released in the future will contain an accurate water mask derived from the same optical imagery. The ArcticDEM represents the height of everything including vegetation and snow on the Earth’s surface (Section 5.2), which also leads to many false positives.

Validation through the crowdsourcing needed more participants because we only received a limited number of responses from the broader community. Possibly, RTSs are a very specific topic, and not many people are interested in or familiar with the topic. In the future we will consider working with instructors who teach courses related to permafrost, remote sensing, or nature hazards and inviting students in their classes as parts of exercise. The system also received some unexpected responses including invalid email addresses and unreliable validation. For example, some people logged in the system using email addresses such as “test@test.com” and “test@gmail.com”, which are invalid after checking. A volunteer validated 14 bounding boxes on 3rd October 2022 by inputting “Yes” as the mapper confidence to all of them, which is unlikely based on the percentage of false positives in the results. The authors checked these 14 inputs and assessed them as unreliable prior to removing them from the database. This indicates that our strategy of validating a bounding box multiple times by different users is necessary. Some people just want to log in the crowdsourcing system and check the functions and capabilities of the system. The system does not require registration for convenience but also allows invalid email addresses. We worried that the crowdsourcing server might not be able to handle large numbers of concurrent users, but this concern was unwarranted. Users could choose among five options: Yes, No, High Confidence, Medium Confidence, and Low Confidence (Section 3.3), but “High”, “Medium”, and “Low” are ambiguous. We may consider only using three options: Yes, No, Unknown for validation in the future.

High-resolution satellite imagery from PlanetScope or Worldview-1, 2, 3, and GeoEye-1 satellites are better datasets for mapping RTSs

(Witharana et al., 2022) in the Arctic and should be considered in the future work. These commercial satellites are too expensive for many users, but some researchers may have access to the imagery through a few channels without paying the cost if they meet certain requirements regarding funding agencies and citizenship, etc.

## 6. Conclusions

We applied feature extraction and deep learning to the ArcticDEM and identified 2494 retrogressive thaw slumps (RTSs) in the Arctic. We set up an online crowdsourcing system that integrated interactive maps of satellite imagery, a window showing composite imagery, and a form to validate the results and recruited volunteers from different channels. The key techniques in the data processing included an automated pipeline, feature extraction, and an object detection algorithm (i.e. YOLOv4). The pipeline also produced intermediate data such as pixel-wise elevation differences, polygons of elevation reductions, lines of narrow-steep slopes, and RTS headwall lines, and these data are potentially useful for other research such as glacier mass balance. The results show that (1) in the scenario where training data are insufficient, the combination of extracted features and deep learning can generate mapping results at a manageable level; (2) the method is efficient and can potentially handle a large dataset and identify RTSs in the Arctic, although the analysis does produce false negatives; (3) the crowdsourcing system is a useful tool and allows for a joint effort from scientific communities but additional outreach and training are needed for identifying RTS; (4) we may identify areas of previously unknown permafrost thaw such as the one in Greenland, indicating the usefulness of our method for monitoring permafrost.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Data availability

ArcticDEM can be download from <http://arcticdem.org>. Data of the extracted features will be available on Arctic Data Center (<https://arcticdata.io/>) after review and technical problems for hosting large datasets (>10 TB) are resolved. Codes for the automated pipeline and ArcticDEM processing are at [https://github.com/yghlc/rs\\_data\\_proc](https://github.com/yghlc/rs_data_proc), YOLOv4 training and prediction are at [https://github.com/yghlc/Landuse\\_DL](https://github.com/yghlc/Landuse_DL). Codes for frontend and back-end of the crowdsourcing system are at <https://github.com/yghlc/labelEarth> and <https://github.com/yghlc/labelEarthServer>, respectively.

### Acknowledgments

Lingcao Huang was supported by the CIRES Visiting Fellows Program, USA, the NOAA Cooperative Agreement, USA (NA17OAR4320101 and NA22OAR4320151) and NSERC PermafrostNet, USA (NETGP 523228–18). Kristy Tiampo was supported by CIRES, USA and the NOAA Cooperative Agreement, USA NA17OAR4320101. The authors would like to thank Lin Liu, Zhuoxuan Xia, Xingyu Xu at the Chinese University of Hong Kong, Enze Zhang at University of Texas Austin, for invaluable suggestions and testing the online crowdsourcing system at its early stage of development. We would like to thank Robert H. Fraser at Canada Centre for Mapping and Earth Observation, Yili Yang at Woodwell Climate Research Center, Ida Rehn at Stockholm University, Eric Wang, Francisco Cerda, Nikhil Raghuraman at Stanford University, and many anonymous volunteers for contributing to the crowdsourcing validation. Great thanks to editors for handling this paper and four anonymous reviewers for their constructive suggestions and insightful comments. ArcticDEM was provided by the Polar Geospatial Center

under NSF-OPP awards 1043681, 1559691, and 1542736. This work utilized the Summit supercomputer, which is supported by the National Science Foundation (awards ACI-1532235 and ACI-1532236), the University of Colorado Boulder, and Colorado State University. The Summit supercomputer is a joint effort of the University of Colorado Boulder and Colorado State University. This work also utilized the Alpine high performance computing resource at the University of Colorado Boulder at its later stage. Alpine is jointly funded by the University of Colorado Boulder, the University of Colorado Anschutz, and Colorado State University.

## References

- Åkerman, H.J., Johansson, M., 2008. Thawing permafrost and thicker active layers in Sub-Arctic Sweden. *Permafr. Periglac. Process.* 19 (3), 279–292.
- Bernhard, P., Zwieback, S., Bergner, N., Hajnsek, I., 2022a. Assessing volumetric change distributions and scaling relations of retrogressive thaw slumps across the Arctic. *Cryosphere* 16 (1), 1–15.
- Bernhard, P., Zwieback, S., Hajnsek, I., 2022b. Accelerated mobilization of organic carbon from retrogressive thaw slumps on the Northern Taymyr Peninsula. *The Cryosphere* 2819–2835.
- Bernhard, P., Zwieback, S., Leinss, S., Hajnsek, I., 2020. Mapping retrogressive thaw slumps using single-pass TanDEM-X observations. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 13, 3263–3280.
- Biskaborn, B.K., Smith, S.L., Noetzi, J., Matthes, H., Vieira, G., Streletskiy, D.A., Schoeneich, P., Romanovsky, V.E., Lewkowicz, A.G., Abramov, A., et al., 2019. Permafrost is warming at a global scale. *Nature Commun.* 10 (1), 264.
- Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
- 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. *Permafr. Periglac. Process.* 25 (4), 243–256.
- Burn, C., Friele, P., 1989. Geomorphology, vegetation succession, soil characteristics and permafrost in retrogressive Thaw Slumps Near Mayo, Yukon Territory. *Arctic* 31–40.
- Burn, C., Lewkowicz, A., 1990. Canadian landform examples-17 retrogressive thaw slumps. *Can. Geogr. (Le Géographe canadien)* 34 (3), 273–276.
- Cable, S., Christiansen, H.H., Westergaard-Nielsen, A., Kroon, A., Elberling, B., 2018. Geomorphological and cryostratigraphical analyses of the Zackenberg Valley, NE Greenland and significance of Holocene alluvial fans. *Geomorphology* 303, 504–523.
- Dai, C., Durand, M., Howat, I.M., Altenau, E.H., Pavelsky, T.M., 2018. Estimating river surface elevation from ArcticDEM. *Geophys. Res. Lett.* 45 (7), 3107–3114.
- Dai, C., Howat, I.M., 2017. Measuring lava flows with ArcticDEM: Application to the 2012–2013 eruption of Tolbachik, Kamchatka. *Geophys. Res. Lett.* 44 (24), 12–133.
- Dijkstra, E.W., et al., 1959. A note on two problems in connexion with graphs. *Numer. Math.* 1 (1), 269–271.
- Godin, E., Fortier, D., Coulombe, S., 2014. Effects of thermo-erosion gully on hydrologic flow networks, discharge and soil loss. *Environ. Res. Lett.* 9 (10), 105010.
- Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. *Deep Learning*, Vol. 1. MIT press Cambridge.
- Hagberg, A., Swart, P., S. Chult, D., 2008. Exploring Network Structure, Dynamics, and Function Using NetworkX. Technical Report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).
- Hjort, J., Karjalainen, O., Aalto, J., Westermann, S., Romanovsky, V.E., Nelson, F.E., Etzelmüller, B., Luoto, M., 2018. Degrading permafrost puts arctic infrastructure at risk by mid-century. *Nature Commun.* 9 (1), 5147.
- Hjort, J., Streletskiy, D., Doré, G., Wu, Q., Bjella, K., Luoto, M., 2022. Impacts of permafrost degradation on infrastructure. *Nat. Rev. Earth Environ.* 3 (1), 24–38.
- 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., Liu, L., 2020. Detecting changes of retrogressive thaw slumps from satellite images using siamese neural network. In: *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, pp. 3090–3093.
- Huang, L., Liu, L., Jiang, L., Zhang, T., 2018. Automatic mapping of thermokarst landforms from remote sensing images using deep learning: A case study in the Northeastern Tibetan Plateau. *Remote Sens.* 10 (12), 2067.
- Huang, L., Liu, L., Luo, J., Lin, Z., Niu, F., 2021. Automatically quantifying evolution of retrogressive thaw slumps in Beiluhe (Tibetan Plateau) from multi-temporal cubesat images. *Int. J. Appl. Earth Obs. Geoinf.* 102, 102399.
- Huang, L., Luo, J., Lin, Z., Niu, F., Liu, L., 2020. Using deep learning to map retrogressive thaw slumps in the Beiluhe Region (Tibetan Plateau) from CubeSat images. *Remote Sens. Environ.* 237, 111534.
- Jorgenson, M.T., 2013. Thermokarst terrains. *Treatise Geomorphol.* 8, 313–324.
- Kokelj, S., Lantz, T., Kanigan, J., Smith, S., Coutts, R., 2009. Origin and polycyclic behaviour of tundra thaw slumps, Mackenzie Delta region, Northwest Territories, Canada. *Permafr. Periglac. Process.* 20 (2), 173–184.
- 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.
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*. Lake Tahoe, NY, USA, pp. 1097–1105.
- Lacelle, D., Brooker, A., Fraser, R.H., Kokelj, S.V., 2015. Distribution and growth of thaw slumps in the Richardson Mountains–Peel Plateau Region, Northwestern Canada. *Geomorphology* 235, 40–51.
- 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).
- Lara, M.J., Chipman, M.L., Hu, F.S., 2019. Automated detection of thermoerosion in permafrost ecosystems using temporally dense landsat image stacks. *Remote Sens. Environ.* 221, 462–473.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521 (7553), 436–444.
- Lewkowicz, A.G., Way, R.G., 2019. Extremes of summer climate trigger thousands of thermokarst landslides in a high arctic environment. *Nature Commun.* 10 (1), 1329.
- Lin, Z., Niu, F., Xu, Z., Xu, J., Wang, P., 2010. Thermal regime of a thermokarst lake and its influence on permafrost, Beiluhe Basin, Qinghai-Tibet Plateau. *Permafr. Periglac. Process.* 21 (4), 315–324.
- Luo, J., Niu, F., Lin, Z., Liu, M., Yin, G., 2019. Recent acceleration of thaw slumping in permafrost terrain of Qinghai-Tibet Plateau: An example from the Beiluhe Region. *Geomorphology*.
- Luo, J., Niu, F., Lin, Z., Liu, M., Yin, G., Gao, Z., 2022. Inventory and frequency of retrogressive thaw slumps in permafrost region of the Qinghai-Tibet Plateau. *Geophys. Res. Lett.* 49 (23), e2022GL099829.
- Miner, K.R., D'Andrilli, J., Mackelprang, R., Edwards, A., Malaska, M.J., Waldrop, M.P., Miller, C.E., 2021. Emergent biogeochemical risks from Arctic permafrost degradation. *Nature Clim. Change* 11 (10), 809–819.
- Miner, K.R., Turetsky, M.R., Malina, E., Bartsch, A., Tamminen, J., McGuire, A.D., Fix, A., Sweeney, C., Elder, C.D., Miller, C.E., 2022. Permafrost carbon emissions in a changing Arctic. *Nat. Rev. Earth Environ.* 3 (1), 55–67.
- Morin, P., Porter, C., Cloutier, M., Howat, I., Noh, M.J., Willis, M., Bates, B., Williamson, C., Peterman, K., 2016. ArcticDEM; a publicly available, high resolution elevation model of the arctic. In: *Egu General Assembly Conference Abstracts*. pp. EPSC2016–8396.
- Mu, C., Abbott, B.W., Norris, A.J., Mu, M., Fan, C., Chen, X., Jia, L., Yang, R., Zhang, T., Wang, K., et al., 2020. The status and stability of permafrost carbon on the Tibetan Plateau. *Earth-Sci. Rev.* 211, 103433.
- Natali, S.M., Holdren, J.P., Rogers, B.M., Treharne, R., Duffy, P.B., Pomeroy, R., MacDonald, E., 2021. Permafrost carbon feedbacks threaten global climate goals. *Proc. Natl. Acad. Sci.* 118 (21), e2100163118.
- Nitze, I., Grosse, G., Jones, B.M., Romanovsky, V.E., Boike, J., 2018. Remote sensing quantifies widespread abundance of permafrost region disturbances across the Arctic and Subarctic. *Nature Commun.* 9 (1), 5423.
- 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.
- Niu, F., Luo, J., Lin, Z., Fang, J., Liu, M., 2016. Thaw-induced slope failures and stability analyses in permafrost regions of the Qinghai-Tibet Plateau, China. *Landslides* 13 (1), 55–65.
- Noh, M.J., Howat, I.M., 2015. Automated stereo-photogrammetric DEM generation at high latitudes: Surface extraction with TIN-based search-space minimization (SETSM) validation and demonstration over glaciated regions. *GISci. Remote Sens.* 52 (2), 198–217.
- Noh, M.J., Howat, I.M., 2017. The surface extraction from TIN based search-space minimization (SETSM) algorithm. *ISPRS J. Photogramm. Remote Sens.* 129, 55–76.
- Nuth, C., Kääb, A., 2011. Co-registration and bias corrections of satellite elevation data sets for quantifying glacier thickness change. *Cryosphere* 5 (1), 271–290.
- Obu, J., Westermann, S., Bartsch, A., Berdnikov, N., Christiansen, H.H., Dashtseren, A., Delaloye, R., Elberling, B., Etzelmüller, B., Kholodov, A., et al., 2019. Northern hemisphere permafrost map based on TTOP modelling for 2000–2016 at 1 km<sup>2</sup> scale. *Earth-Sci. Rev.* 193, 299–316.
- Pang, Q., Zhao, L., Li, S., Ding, Y., 2012. Active layer thickness variations on the Qinghai-Tibet Plateau under the scenarios of climate change. *Environ. Earth Sci.* 66 (3), 849–857.
- Pastor, A., Skovsholt, L.J., Christoffersen, K.S., Wu, N., Riis, T., 2021. Geomorphology and vegetation drive hydrochemistry changes in two Northeast Greenland streams. *Hydrol. Process.* 35 (10), e14369.
- Pekel, J.F., Cottam, A., Gorelick, N., Belward, A.S., 2016. High-resolution mapping of global surface water and its long-term changes. *Nature* 540 (7633), 418–422.
- Porter, C., Morin, P., Howat, I., Noh, M.J., Bates, B., Peterman, K., Keese, S., Schlenk, M., Gardiner, J., Tomko, K., Willis, M., Kelleher, C., Cloutier, M., Husby, E., Foga, S., Nakamura, H., Platson, M., Wethington, Jr., M., Williamson, C., Bauer, G., Enos, J., Arnold, G., Kramer, W., Becker, P., Doshi, A., D'Souza, C., Cummins, P., Laurier, F., Bojesen, M., 2018. ArcticDEM. Harvard Dataverse, <http://dx.doi.org/10.7910/DVN/OHHUKH>, (Accessed 7 March 2023).

- Ramage, J.L., Irrgang, A.M., Herzsich, U., Morgenstern, A., Couture, N., Lantuit, H., 2017. Terrain controls on the occurrence of coastal retrogressive thaw slumps along the Yukon Coast, Canada. *J. Geophys. Res. Earth Surf.* 122 (9), 1619–1634.
- Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 779–788.
- Rudy, A.C., Lamoureux, S.F., Treitz, P., Collingwood, A., 2013. Identifying permafrost slope disturbance using multi-temporal optical satellite images and change detection techniques. *Cold Reg. Sci. & Technol.* 88, 37–49.
- Runge, A., Nitze, I., Grosse, G., 2022. Remote sensing annual dynamics of rapid permafrost thaw disturbances with LandTrendr. *Remote Sens. Environ.* 268, 112752.
- Schaefer, K., Elshorbany, Y., Jafarov, E., Schuster, P.F., Striegl, R.G., Wickland, K.P., Sunderland, E.M., 2020. Potential impacts of mercury released from thawing permafrost. *Nat. Commun.* 11 (1), 1–6.
- Schuur, E.A.G., McGuire, A.D., Schdel, C., Grosse, G., Harden, J.W., Hayes, D.J., Hugelius, G., Koven, C.D., Kuhry, P., Lawrence, D.M., 2015. Climate change and the permafrost carbon feedback. *Nature* 520 (7546), 171–179.
- Smith, S.L., O'Neill, H.B., Isaksen, K., Noetzi, J., Romanovsky, V.E., 2022. The changing thermal state of permafrost. *Nat. Rev. Earth Environ.* 3 (1), 10–23.
- Steedman, A.E., Lantz, T.C., Kokelj, S.V., 2017. Spatio-temporal variation in high-centre polygons and ice-wedge melt ponds, Tuktoyaktuk coastlands, Northwest Territories. *Permafr. Periglac. Process.* 28 (1), 66–78.
- Swanson, D.K., 2021. Permafrost thaw-related slope failures in Alaska's Arctic National Parks, c. 1980–2019. *Permafr. Periglac. Process.* 32 (3), 392–406.
- Tarnocai, C., Canadell, J.G., Schuur, E.A.G., Kuhry, P., Mazhitova, G., Zimov, S., 2009. Soil organic carbon pools in the northern Circumpolar Permafrost Region. *Glob. Biogeochem. Cycles* 23 (2).
- Teufel, B., Sushama, L., 2019. Abrupt changes across the Arctic Permafrost Region endanger northern development. *Nature Clim. Change* 9 (11), 858–862.
- 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.
- van der Sluijs, J., Kokelj, S.V., Tunnicliffe, J.F., 2022. Allometric scaling of retrogressive thaw slumps. *Cryosphere Discuss.* 1–30.
- Van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Goullart, E., Yu, T., 2014. Scikit-image: image processing in Python. *PeerJ* 2, e453. <http://dx.doi.org/10.7717/peerj.453>.
- Vedaldi, A., Soatto, S., 2008. Quick shift and kernel methods for mode seeking. In: *European Conference on Computer Vision*. Springer, pp. 705–718.
- Wang, B., Paudel, B., Li, H., 2009. Retrogression characteristics of landslides in fine-grained permafrost soils, Mackenzie Valley, Canada. *Landslides* 6 (2), 121–127.
- Ward Jones, M.K., 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.
- 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.
- Xia, Z., Huang, L., Fan, C., Jia, S., Lin, Z., Liu, L., Luo, J., Niu, F., Zhang, T., 2022. Retrogressive thaw slumps along the Qinghai-Tibet engineering corridor: A comprehensive inventory and their distribution characteristics. *Earth System Science Data* 14 (9), 3875–3887.
- Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2021. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* 64 (3), 107–115.
- Zimov, S.A., Schuur, E.A., Chapin, III, F.S., 2006. Permafrost and the global carbon budget. *Science* 312 (5780), 1612–1613.