Irregular Spot Detection and Tracking: Automatically Geo-Temporal Tracking of Supra-Glacial Lakes on the Greenland Ice Sheet

Yu-Li Liang [1], Qin Lv [2]

[1]Yu-Li.Liang@colorado.edu [2]Qin.Lv@colorado.edu,

Department of Computer Science
Technical Report
CU-CS 1078-11
University of Colorado at Boulder

Preprint submitted to Computer Science Technical Report *February 15, 2011*

Abstract

Supra-glacial lakes (i.e., ponds of melting water on ice sheet) in Greenland have attracted extensive global attention during the recent years. To understand the important role they play in glacier movement, sea level rise, and climate change, scientists need to learn where these lakes are, when they form, and how they change in each melting season and across multiple years. This requires detecting and tracking supra-glacial lakes both spatially and temporally. This problem is challenging due to the diverse qualities of massive amount of remote sensing images, frequent cloud coverage, as well as the diversity and dynamics of the large number of supra-glacial lakes on the Greenland ice sheet. Previous works that use supervised methods to detect supra-glacial lakes in individual cloud-free satellite images are limited in scale, quality, and functionality.

With supra-glacial lakes are shown as "spots" in images, we propose an effective solution to automatically detect and track time-varying spots from cloudy time-series images. In other words, this framework could be applied to any other problems to automatically detect or track irregular-shape spots from noisy images. Specifically, we propose novel techniques to (1) Select images: select the best-quality image within each time interval; (2) Spot detection: using adaptive thresholding to detect supra-glacial lakes in individual images with diverse quality; and (3) Spot tracking: track lakes across time series of images as lakes appear, change in size, merge or split, and disappear. The proposed solution has been evaluated using 10 years of MODIS data (i.e., 2000 to 2009). The results demonstrate that our proposed solution can automatically detect and track supra-glacial lakes with high efficiency and high accuracy: 96.3% tracked lakes are Supra-glacial lakes (precision is 0.963 over 1.0), and 99.0% supra-glacial lakes could be found by our tracking algorithm (recall is 0.990 over 1.0).

1. Introduction

The supra-glacial lakes in Greenland ice sheet play important roles in sea level rise and global environment change.Recently, the increased glacial movement has resulted in outlet glaciers discharging more ice directly into the ocean and causing

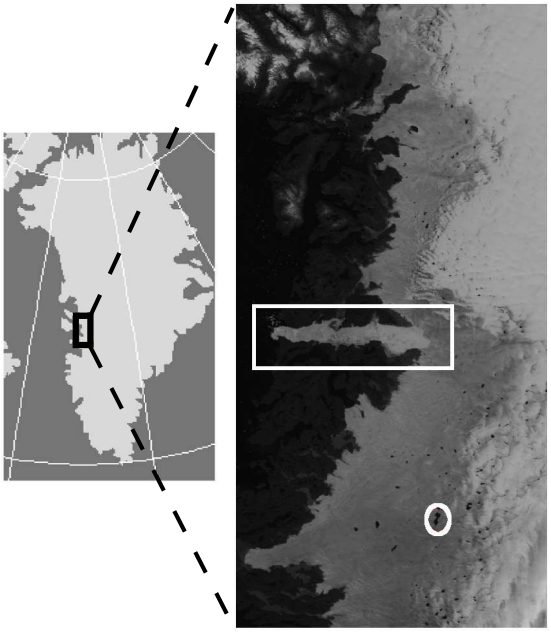

Figure 1: Left: Our target region (black rectangular) in Greenland. Right: A satellite image on July 8th, 2009 – light gray and white area on the east is ice sheet; dark gray and black region on the west is land and sea; the white rectangle indicates Jakobshavn Isbrae, a large outlet glacier which flows from east (ice sheet) to west (sea); and the small white circle indicates one supra-glacial lake.

sea level rise, which has significant environmental and social impacts all over the world. The motion of the Greenland ice sheet, especially its outlet glaciers, is closely related to supra-glacial lakes, which are ponds of melting water on top of the glaciers. Supra-glacial lakes are good indicators of the ice sheet's melting status. One famous hypothesis from Zwally et al. (2002) is drainage of supra-glacial lakes reaches the base of the ice sheet, lubricates the surface, and creates a mechanism for faster glacier motion. This hypothesis is still in debate. Many previous studies, such as Joughin et al. (2008); Das et al. (2008); Bartholomew et al. (2010), have tried to test this hypothesis by studying the temporal variation of supra-glacial lakes in small regions with 1, 2, or 4 lakes. Detecting and tracking supra-glacial lakes over a much wider geographical region and longer time periods is extremely desirable, but has never been achieved before.

How to automate geo-temporal tracking supra-glacial lakes is an important topic. Most studies rely on manual lake identification and it is time consuming and infeasible for large-scale investigation. Figure 1 shows a our target region, which

is centered at Jakobshavn Isbrae and includes around $16,500\ km^2$ of ice sheet. On every day during melting season, there are up to 200 lakes in this target region. Assume there are 100 lakes in each day and summer is around 150 days (i.e., May 1st Oct. 1st), one would need to manually identify $100 \times 150 = 15,000$ lakes while studying lake variations in this area. And this number is only for one single year. Therefore, manual tracking is extramely impractical. Instead, computer-aided automatic geo-temporal tracking of supra-glacial lakes is required.

There is reason why people have talked about "automatically geo-temporal tracking lakes" for years but it has never been done until this work. It poses several unique challenges:

- *A huge amount of satellite image data* need to be processed in order to detect and track supra-glacier lakes in different geographical locations and temporal periods.

- *A large number of diverse and dynamic supra-glacial lakes* need to be identified and tracked from *inhomogeneous background*. All lakes are different in size, shape, depth (i.e. reflectance), etc., and may appear, disappear, merge, or split over time. At the same time, reflectance of ice vary with humidity also worsen the problem.

- *Heterogeneous satellite image qualities* exist, due to the difference in satellite viewing angle, solar angle, weather condition, and equipment conditions. Images could be sharp or blur, bright or dark, cloudy or clear, and sometimes with unexpected missing signal. Especially, cloud is difficult to distinguish from ice while cloud coverage largely affect lake appearance in the satellite images. Those factors all pose serious challenge in detecting lakes.

Since lakes appear as prominent spots above ice sheet (Figure 1), we propose an effective and efficient solution for tracking irregular-shape spots using time series of images. Targeting the specific challenges discussed above, our solution exploits a novel three-stage temporal image analysis process that consists of the following:

- Image selection: Select the best-quality satellite image within each time period (e.g., a day);

- Spot Detection: Detect supra-glacial lakes in individual images using adaptive thresholding; and

- Spot Tracking: Temporal track supra-glacial lakes using time series of satellite images.

We evaluate our solution by using real satellite images over multiple years. Detailed results shown in Section 7 demonstrate that our proposed solution achieves high accuracy and high efficiency.

The proposed solution could be widely applied in many other scenarios. "Automatically tracking irregular-shape spots from cloudy images" is actually the solution for remote sensing surface change, such as estimating oil spill, forest infection of pine beetle, etc. Furthermore, "detect irregular spot from inhomogeneous background" is an even more fundamental properties and could be applied in various application, such as facial spot detection, leaf infection estimation, etc. For readers who are interested in applying our technique to other problems, could take a look at Section 8, where we briefly summarize the properties of our algorithm and compare it with previous works.

The remainder of this article is organized as follows. Section 2 presents the problem formulation and gives an overview of the proposed solution. Sections 3, 4, 5 and 6 present in detail the proposed techniques. Detailed evaluation results are discussed in Section 7. We briefly summarize our algorithm, possible application, and discuss related works in Section 8. Conclusion is in Section 9.

2. Problem Formulation and Design Overview

Figure 2 gives an overview of our design and illustrates the key components in the proposed solution. Black box with bold solid line enclose our solution. Above the black box is input data and pre-processing method. Below the black box is output result. In this section, we focus on a brief overview of our design. Detailed specifications could be found in Section 7.1.

We use Moderate Resolution Imaging Spectroradiometer (MODIS) surface reflectance time series data (`http://modis.gsfc.nasa.gov/`), or *MODIS data* in Figure 2, as our raw data. The reason is that *MODIS data* are in sufficient spatial resoultion (i.e., 250 meter) and temporal resolution (i.e., daily data). In this work, we are interested in the supra-glacial lakes on the ice sheet near Jakobshavn Isbrae, which is a large outlet glacier in west Greenland (Figure 1). Since different images were taken in different viewing angles, we first map or grid all the MODIS images into the same coordinate system (e.g., EASE-Grid Overview: `http://nsidc.org/data/ease/`). After this process, all mapped images corresponding to the same coordinates and the same region.

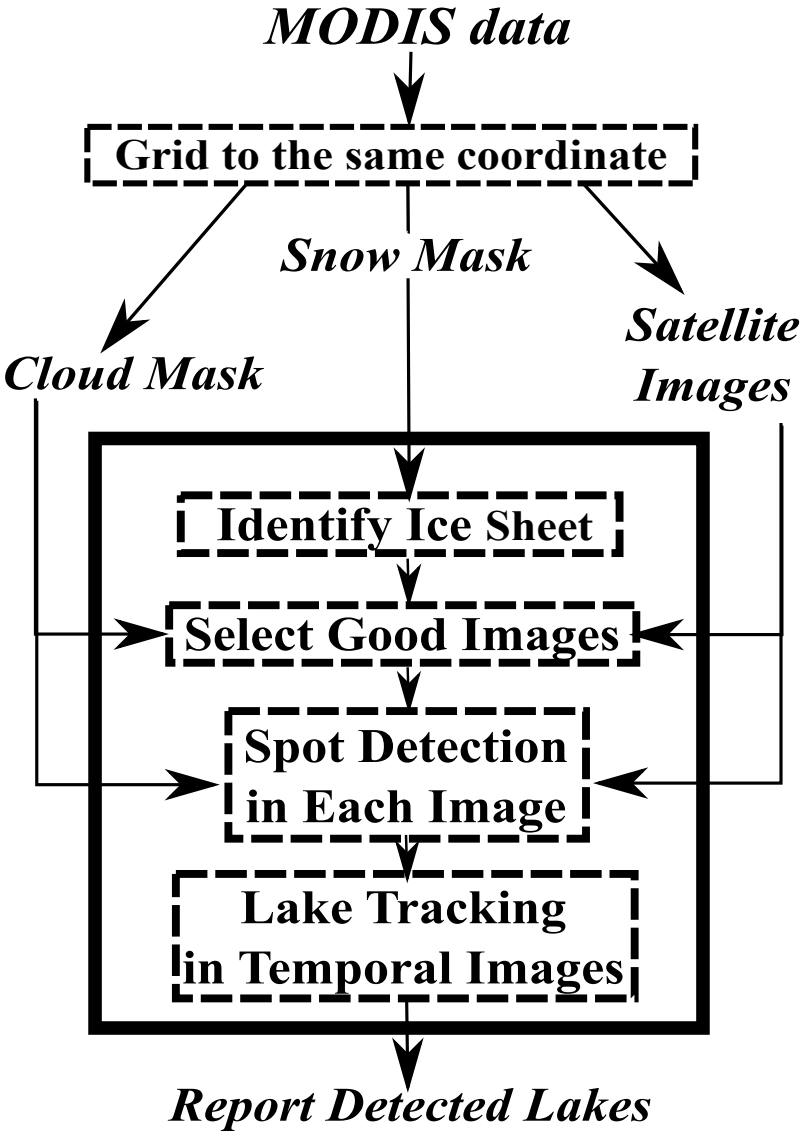

Figure 2: Design overview of Geo-temporal tracking of supra-glacial lakes. Black box with bold solid line: Our solution; Above the black box: Preprocess and input data; Below the black box: Output result.

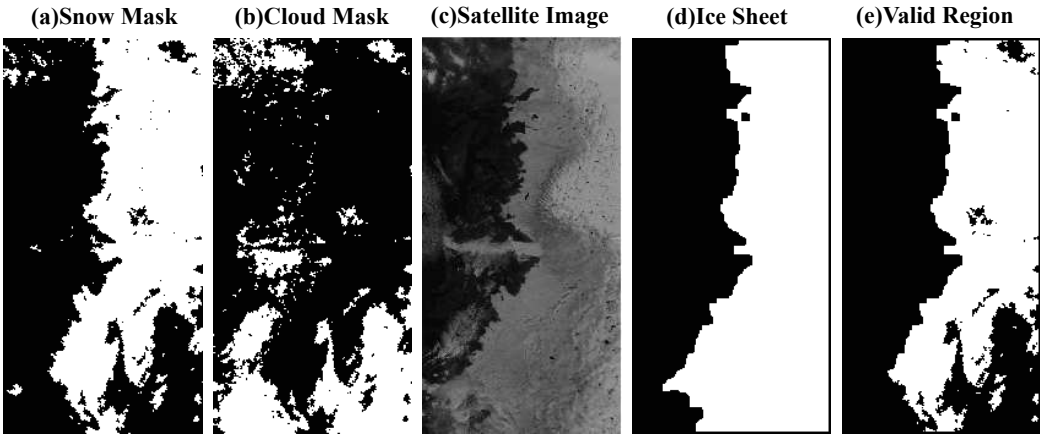

Figure 3: Three input images (a)–(c) and the identified ice sheet (d)–(e) for July 30th, 2006. (a) Snow Mask: Positive (white) pixels are snow-covered land. (b) Could Mask: Positive (white) pixels are cloud. (c) Satellite Image: Main image data that contain lake information. (d) Ice Sheet: Positive (white) pixels are major ice sheet. These pixels are automatically determined by our method. (e) Valid Region: Positive (white) pixels are cloud-free snow-covered land, where we look for potential lakes.

There are three type of mapped images we used as input: *Satellite Images*, *Snow Masks*, and *Cloud Masks*. *Satellite Images* is surface reflectance images of target area, such as figure Figure 3-(c). *Snow Masks* indicate cloud-free snow-covered land of each *Satellite Image* (i.e., Figure 3-(a)). *Cloud Masks* flag pixels that are obscured by cloud in *Satellite Images*. But cloud mask is not accurate all the time, such as Figure 3-(b). In the following section, "images" refer to the *Satellite Images* unless otherwise specified.

Black box with bold solid line in Figure 2 enclose our proposed algorithm. We first identify the ice sheet and specific region where lakes appear (Section 3). Next, for each time interval (e.g., daily), we examine all available images within that time interval and select the highest-quality one for futher analysis (Section 4). We then propose novel techniques for spot (i.e., lake) detection in each single image (Section 5). Finally, lake tracking across a series of temporal images was presented in Section 6.

The end results record interested properties of geo-temporal tracked supra-lakes, such as area variation, disappearing day, etc. In summary, our solution is the first of its kind that automatically detects and tracks supra-glacial lakes in a time series of (potentially) cloudy satellite images in the west Greenland.

3. Determine Analysis Area: Valid Region

The *valid region* is where we look for supra-glacial lakes. As shown in Figure 1, satellite images contain not only ice sheet, but also land and sea areas. The ice sheet could also be obscured by cloud or with corrupted signals (e.g., Figure 3-(c)). Therefore, we need to determine the valid region, which corresponds to cloud-free ice sheet pixels with valid signals.

First, we identify where the major ice sheet is. The snow mask from the Snow Cover data (Section 7.1) indicates pixels of ice sheet with reasonably good accuracy (Figure 3-(a)). However, it sometimes excludes supra-glacial lakes, mistakenly treats cloud as snow-covered land, or includes snow-covered area which is not permanent ice sheet. All those inaccuracy reduce the quality of later analysis. To address this problem, we generate one *ice sheet* for each year, and the ice sheet is composed of pixels which are snow-covered in the snow masks more than half of the summer season. We then apply several rounds of morphology operation (i.e., erosion and dilation) to effectively remove small glitches. If there are more than one piece of ice sheet, we calculate the area of each ice sheet and pick the largest one. On example of the calculated ice sheet (for Year 2006) is shown in Figure 3-(d).

Next, we consider the cloud mask information. For each image, we derive the corresponding cloud mask from the Snow Cover data (Section 7.1), which shows cloud-obscured pixels. Again, we apply several rounds of morphology operation (i.e., erosion and dilation) on the cloud masks to remove tiny granules and seal small holes.

We can now determine the valid region using the processed ice sheet and cloud mask information. Specifically, the valid region is composed of pixels which are (1) snow-covered in the ice sheet, (2) not obscured by cloud in the cloud mask, and (3) have valid signals in the satellite image (Figure 3-(c)). Only pixels in the valid region have a value of 1 in the valid region map VM:

$$\begin{aligned} VM(r,c) = 1: &\quad \text{valid signal in ice sheet without cloud} \\ = 0: &\quad \text{otherwise} \end{aligned} \tag{1}$$

4. Select Best-Quality Image

We propose to select one best-quality image in each day. Generally, in our target region, multiple satellite images may have been collected in each day. These images were captured under different situations and may vary in satellite angle,

solar angle, weather condition, equipment condition, etc. Therefore, the qualities of these images such as sharpness, brightness, cloud coverage, etc., may also vary. Using all these images may cause unnecessary confusion and noise in later analysis. Therefore, we choose only one best-quality image per day when there are good image available.

We extract "quality" directly from the image content instead of solar angle (i.e., bright or dark) and satellite angle (i.e., sharp or blur image). For example, we estimate an image is blur or not based on reflectance signals instead of its satellite viewing angle. One reason is that retrieveing extra physical properties takes extra effort in collecting and preprocessing MODIS data (i.e., MOD03), which is extremly time-consuming and we undesirable. Anohter reason is cloud coverage. For example, if a sharp and bright image mostly covered by cloud, it is still useless for us. In other words, knowing solar angle and satellite angle would not quarantee us a "good" image while it take lots time.

In here, good quality means (1) less cloudy, (2) brighter, (3) shaper, and (4) containing lakes on the ice sheet. We define four coefficients, which correspond to those properties:

1. Less cloudy = larger *Effective Area*
2. Brighter = higher *Average Intensity*
3. Sharper = larger *Sharpness Coefficient*
4. Containing lakes = *Percentage of Dark Spot Pixels* is larger than zero.

For each image, the four coefficients are retrieved as following. Assume there are K satellite images for each day. Let F_k be the k-th satellite image ($k \in [1, K]$). Using the valid region and valid region map we have determined above (Equation 1), we have

- Effective Area E_k: Number of pixels in the valid region. Images with larger Effective Area is less cloudy in our target region, therefore more desirable.

$$E_k = \sum_{r=1}^{R} \sum_{c=1}^{C} VM_k(r, c)$$

- Average Intensity $\bar{I}_k$: Mean of all pixels' intensity values in the valid region. In here, "intensity" of images is reflectance in satellite images. Larger $\bar{I}_k$ indicates brighter image, and dark spots (lakes) are easier to detect in

brighter images.

$$\bar{I}_k = \frac{1}{E_k} \sum_{r=1}^{R} \sum_{c=1}^{C} F_k(r,c) \times VM_k(r,c)$$

- Sharpness Coefficient S_k: Sum of the first derivative along row and the first derivative along column for all pixels in the valid region, divided by Average Intensity.

$$S_k = \sum_{VM_k(r,c)==1} \big\{ (F_k(r,c) - F_k(r-1,c)) + (F_k(r,c) - F_k(r,c-1)) \big\} / \bar{I}_k \tag{2}$$

- Percentage of Dark Spot Pixels P_k^{SP}: Let N_k^{SP} be the number of dark spot pixels in the valid region, as calculated in Section 5.1 and 5.2, then P_k^{SP} is the proportion of dark spot pixels to Effective Area: This is needed to ensure that there are dark spots (possibly lakes) in the selected image.

$$P_k^{SP} = N_k^{SP} / E_k$$

Within each day, we collect qualified images which are (1) not so cloudy, (2) not so dark, and (4) containing lakes on the ice sheet. In other words, an *qualified* image satisfies the three criteria: (1) $E_k > E_{thr}$, (2) $\bar{I}_k > \bar{I}_{thr}$, and (3) $P_k^{SP} > 0$. As listed in Table 1, E_{thr} and $\bar{I}_{thr}$ are the minimum effective area and minimum average intensity of qualified images. We ignore the days which contains no qualified image, since including that day would only introduce noise.

We then choose the best-quality image among the qualified images of that day. The best-quality image is the one with the largest *Quality Coefficient* Q_k:

$$\begin{aligned} Q_k =& E_k/max(E_u) + \bar{I}_k/max(\bar{I}_u) \\ &+ S_k/max(S_u) + P_k^{SP}/max(P_u^{SP}) \\ &\text{for } u = 1, ..., K \end{aligned} \tag{3}$$

In the following sections, unless otherwise specified, the images are best-quality images selected from each day.

5. Lake Detection in Single Image

The key challenge of lake detection is identify a feature that is stable among images with diverse qualities. The images we use are MODIS data with 250-meter

resolution, which contains reflectance in two bands (visible red and infrared). In both bands, lake reflectance (i.e., water) is generally smaller than that of ice. Previous approaches by Box & Ski (2007) differentiated lakes and ice sheet in a small area by finding *one fixed threshold* in each or both reflectance bands.

We found that there is *NO* one fixed reflectance threshold could successfully classify lakes across time (e.g., 10 years) and space (e.g., target region in Figure 1). A fixed threshold does not work across time because reflectance of a geographical feature changes with the solar zenith angle (SZA) and satellite viewing angle (SVA). Although it is possible to select images with the same SZA and SVA, since the repeat cycle of MODIS satellite is 16 days, two images with identical SZA and SVA are at least 16 days apart. Such temporal resolution is insufficient for tracking lakes, which change in days or drain overnight. Also, a fixed threshold does not work across space because of the variation of ice humidity. As a result, one fixed reflectance threshold could easily misclassify a shallow lake (with mid reflectance) as ice or a large piece of wet ice (with low reflectance) as lake.

We propose to detect lakes using the most basic fact: *Reflectance of each lake is smaller than that of the ice surrounding the lake*. Human eyes could easily identify supra-glacial lakes as darker spots surrounded by whiter ice sheet (Figure 1). So the robust feature we are looking for is *spots with high contrast with its surrounding area*. This property is true in both band 1 reflectance (red) and band 2 reflectance (infrared). Since land is more different from ice sheet in band 1, we choose to use only band 1 reflectance (optical wavelength: 620nm to 670 nm) in our experiments. We further search the threshold among those "contrast" values dynamically to accommodate diversity of image qualities.

5.1. Calculate Contrast Coefficient

Our chosen feature is *spots with high contrast with its surroundings*. We calculate the "contrast" of each interested region and its neighborhood. The *Structuring Element* (SE) pair, SE_{in} and SE_{out} (Figure 4) is used to facilitate this operation. Specifically, (1)SE_{in} is for generating the average intensity of every interested region, (2) SE_{out} is used to generate the average intensity of the neighborhood of every interested region, and (3) "contrast" is intensity difference of every interested region and its neighborhood. Let *Layer-L* be the elements that are L pixels away from the center. The value of each pixel in the L-th layer is W_L:

$$W_L = 1/(L \times 8). \tag{4}$$

We then calculate the *contrast coefficient* based on the intensity difference of interested region and its neighborhood. The size of SE_{in} denotes the size of

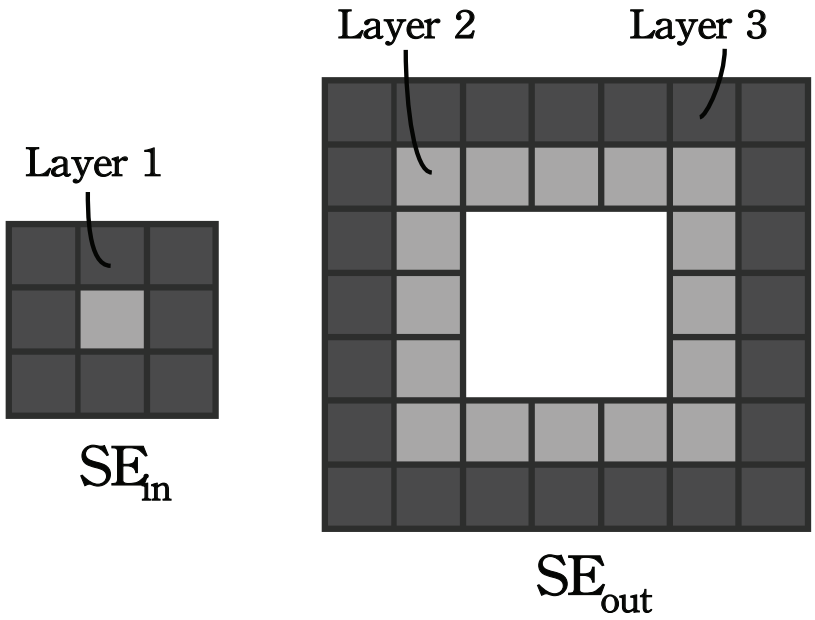

Figure 4: Structuring Element (SE) pair: SE pair is used to calculate the contrast coefficient, which is the intensity difference of interested region and its neighborhood. (Left) a 3×3 inner SE (SE_{in}) with 1 center pixel (light gray) and layer-1 pixels (dark gray). SE_{in} is used to calculate the average intensity of interested region. (Right) a 7×7 outer SE (SE_{out}) with layer-2 (light gray) and layer-3 pixels (dark gray). SE_{out} is used to calculate the average intensity of neighborhood. The actual sizes of SE pairs we used in experiments are listed in Table 1.

interested region and corresponds to the minimum size of lakes in our case. The size of SE_{out} is neighborhood of the interested region and corresponds to the maximum size of lakes. Let W_{in} and W_{out} be the sum of all the element values in SE_{in} and SE_{out}, respectively. Let F be an image (with reflectance values) and $\otimes$ be 2-D convolution operation. Then the average intensity of each interested region is $(F \otimes SE_{in})/W_{in}$ and the average intensity of its neighborhood is $(F \otimes SE_{out})/W_{out}$. And the contrast coefficients (CC) is computed as following:

$$CC = (F \otimes SE_{out})/W_{out} - (F \otimes SE_{in})/W_{in} \tag{5}$$

The distribution of all contract coefficient of an image could be shown in a histogram, like Figure 5. Number of histogram bins (N_{bin}) heavily affect the shape of histogram, which affect dynamic threshold search (Section 5.2). Therefore, N_{bin} need to be an appropriate magnitude. And, since we compare images within each year to search lakes, N_{bin} is fixed within each year but adjustable to yearly

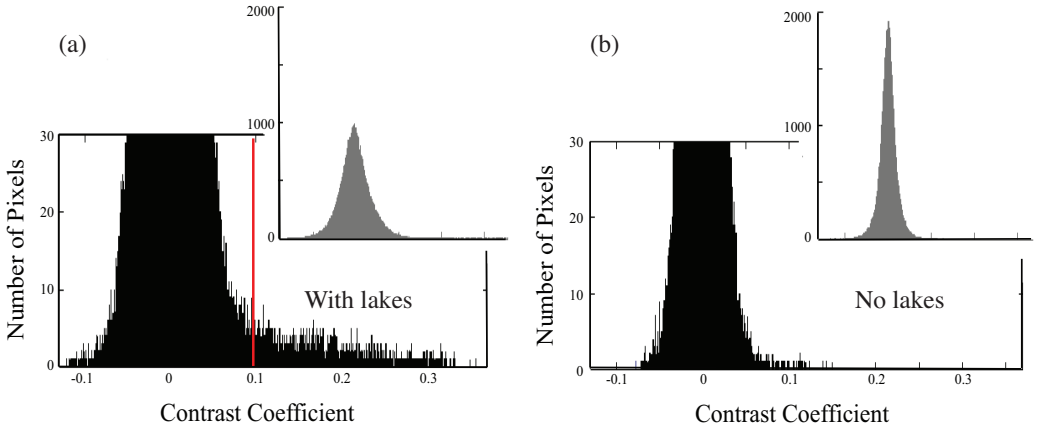

Figure 5: Contrast Coefficient (CC) histograms of images with and without lakes. In both (a) and (b), the gray plots on the right are the overall histograms. Black plots are the zoom-in view of the bottom of overall histograms. (a) For images with lakes, CC histogram has a thin tail on the right with high contrast (i.e., spots or lakes on ice sheet). Our dynamic thresholding algorithm can determine the proper threshold (the red vertical line in (a)) to best separate the big bump (background pixels) and the tail (lakes). (b) For images without lakes, there is no such tail in the CC histogram.

change. Assume IS is the ice sheet (Figure 3-(d), Section 3), which is a binary map and indicate where is ice, N_{bin} is defined as:

$$N_{bin} = \sum_{r=1}^{R}\sum_{c=1}^{C} IS(r,c) \Big/ \sqrt[3]{\sum_{r=1}^{R}\sum_{c=1}^{C} IS(r,c)} \tag{6}$$

5.2. Determine Contrast Threshold

As shown in Figure 5, histograms with and without lakes have distinct features. Specifically, the histogram of the no-lake image follows a symmetric Gaussian-like distribution, which approximately centers at zero (Figure 5-(b)), while the histogram of the image with lakes composed of a big bump, which is a zero-center Gaussian-like distribution, and a thin tail on the right-hand side. right (Figure 5-(a)). The morphology of thin tail is various and could be a tapering tail, a tiny bump, or a non-continuous tail. The big bump corresponds to the background ice sheet (low contrast), while the rest portion is the pixels of lakes (prominent dark spots with high contrast). For images with different background intensity distributions, the shape of the histogram may change correspondingly but maintain the distinct big-bump and thin-tail parts.

For images with lakes, a good contrast threshold could properly separate the dark spot pixels (thin tail) and the background pixels (major bump). Because of the wide variation of satellite images, contrast threshold is not a fixed value and has to be dynamically adjusted in an automatic fashion. Choosing a threshold proportional to the standard deviation of histogram is a popular choice, such as works by Dengler et al. (1993); Olivo-Marin (2002). However, due to the diversity of background intensity in the satellite images, this method does not work well.

We automatically determine the contrast threshold and prominant spots based on feature of histogrm shpae. Histogram with lakes is composed of a large bump and a thin right-hand tail (Figure 5-(a)). Regardless the shape of thin tail, the 1^{st} derivative of histogram is close to zero between big bump and tiny tail. Therefore, we define the threshold as "the smallest close-to-zero 1^{st} derivative on the right-hand side of big bump". Because histogram are very spiky, before threshold searching, we smooth histogram to stablize result. We apply *Moving Average* to smooth histogram (`http://en.wikipedia.org/wiki/Moving_average`). A *window* include a subset of histogram and calculate the average of this subset. By rolling this *window* from beginning to the end of histogram, we could generate a series of "averages of different subsets", which is a smoother version of origianl data set. Figure 7.5 shows an example: histogram (i.e., black),

Moving Average in a smaller window (i.e., red soild line), and Moving Average result by a larger window (i.e., red dashed line).

We automatically determine the contrast threshold in each image. Since we are looking for "dark" spot, the following steps are only applied on the histogram with larger-than-zero contrast coefficient:

1. Smooth histogram: using the Moving Average (with window size equal to $N_{bin}/X1$ bins of histogram).
2. Calculate the 1^{st} derivative: It means the 1^{st} derivative of previously smoothed histogram with respect to contrast coefficient.
3. Smooth 1^{st} derivative curve: Apply Moving Average again to the smoothed 1^{st} derivative curve (with window size equal to $N_{bin}/X2$ bins of histogram).
4. Define threshold: Threshold is the smallest contrast coefficient where smoothed 1^{st} derivative is equal to $X3$.
5. Extract spots: Pixels with higher-than-threshold contrast coefficients are prominant spots.

The role of three varilabes (i.e., $X1$, $X2$, and $X3$) used above are descrubed below. $X1$ and $X2$ are both related to Moving Average window which stablize result. Therefore, the values of those two should be (1) not too large to result in noisy curve, and (2) not too small and result in an unfaithful curve (Figure 7.5). On the other hand, $X3$ works like a threshold in the above procedure. Smaller $X3$ results in larger threshold and reported spots are only dark pixels, Larger $X3$ results in smller threshold and spots include more gray pixels, such as rim of a lake or localized wet ice. Therefore, chosen value of $X3$ depends on the user preference. For these variables, the values we used are listed in Table 2 and sensitivity analysis is presented in Section 7.5.

5.3. Refine Detected Spots

We automatically determine prominant spots in Section 5.2 and further refinements are needed. First, we refine contrast coefficients. When calculating the contrast coefficients, "neighborhood" may contain both dark pixels (lakes) and bright pixels (ice sheet). For better accuracy, we should include only bright background pixels (ice sheet) while calculating average intensity of neighborhood. The process works as follows:

- For a given image F, we use the methods in Section 5.1 and Section 5.2 to generate one set of spot pixels $S1$. We represent $S1$ as a binary map, in which only detected dark spot pixels have a value of 1.

- Repeat Section 5.1 and Section 5.2 to generate spot pixels $S2$. But this time, we correct the contrast coefficient by only include non-spot pixels in SE_{out}. Let N be a binary map, in which a pixel is set to 1 only if it is in the valid region and is not a spot pixel according to $S1$. We then generate a corresponding image $F' = F \otimes N$. Similar to Equation 5, the more accurate contrast coefficient CC' is:

$$CC' = (F' \otimes SE_{out})/(N \otimes SE_{out}) - (F' \otimes SE_{in})/W_{in}$$

Second, we remove spots with inappropriate size. We eliminate spots contains only one pixel because they are very likely be noise. In other words, the smallest lake we consider is 2 pixels with area equal to 0.125 Km^2. Besides, we also remove the spots which their dimension is too large. Our spot detection algorithm flag not only pixels in round spots, but also long prominent lines. Assume O_{SE} (Table 1) as the maximum dimension (height or width) of desirable spots, the magnitude is defined by users according to different application. We prune out the spots with dimension exceeding O_{SE}.

6. Lake Tracking in Time-series Images

Tracking supra-glacial lakes in a serious cloudy images poses several challenges. For supra-glacial lakes, lake shape and depth change with time and only location is relatively stable. Besides, frequent cloud screen worsens the problem. Since physical properties of ice and cloud are similar, it is difficult to detect cloud above ice. The snow mask from Snow Cover data (MOD10-L2 in Section 7.1) is not always accurate. However, lakes could be missed because of cloud, which shows frequently in satellite images. Therefore, when we do not see a lake in a given image, we are not sure if the lake actually disappears or it is just obscured by cloud.

To address aforementioned issues, we propose a lake tracking method that exploits the temporal consistency of lakes. Specifically, lakes persist for days, while clouds and noises change shape and location in minutes. For example, a dark spot which appeared only once could be a lake or noise. However, if it shows consistently in multiple days, it is very likely a lake. Before we explain our temporal tracking algorithm in detail, we need to define four terms:

- *Next image*: Since we choose one best-quality image for each day, the *next image* refers to the image from the next available day. For example, if the

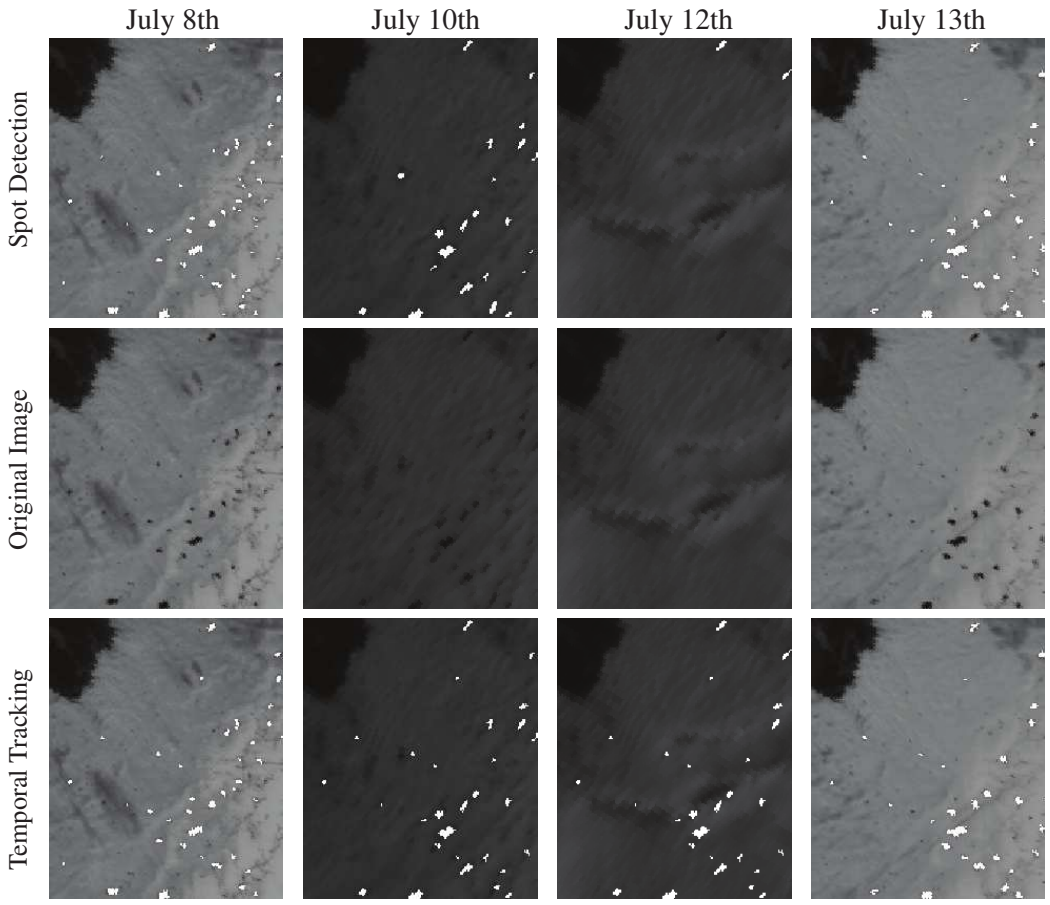

Figure 6: Results of spot detection (row 1), original satellite images (row 2), and temporal tracking (row 3). Brightest white pixels are detected spots. Each column contains images of each day. From left to right is July 8th, 10th, 12th, and 13th in Year 2006. Spot detection works well in cloud-free images (July 8th and 13th), but not for less clear and cloudy images (July 10th and 12th). Results of temporal tracking are much more robust (row 3). On average, temporal tracking removes 21.4% of spots which are noise (i.e., non-lake dark spots) and recovers 43.5% of spots which are unseen lakes (Section 7.4).

current image is from July 1st, 2003, then its next image is the best-quality image on July 2nd, 2003. If there is no qualified image on July 2nd, the next image would be the best-quality image on July 3rd, and so on. Similarly for previous image.

- *Mapping*: When we say one spot in the current image *maps* to another spot in the next image, geographical location of two spots overlap with each other fully or partially.

- *Corresponding spot*: To track a lake across time, we need to find its *corresponding spot(s)* in a time-series of images. *Corresponding spot(s)* should be similar with each other and is not necessary to be exactly the same. Assume there is a spot x in day D_x and spot(s) y in a later day D_y. If y and x represent the same lake, both should be similar in location and area. Assume area of x and y is x_{area} and y_{area} respectively. We define y as the corresponding spot(s) of x (i.e., register y to x) if:

  - y is mapped by x;
  - $y_{area} \leq 2 \times x_{area} + 8$; and
  - $y_{area} > 0.5 \times x_{area}$.

- *True miss*: We define that a lake has one *true miss* when it is unseen one time in the cloud-free region, based on cloud mask. However, since cloud mask is not very reliable, the *true miss* could resulted from cloud screen, while it is rare. Because cloud mask did report cloud, expecially large chunk of cloud. Since cloud changes in transient, it is very rare for lakes to be covered by small undetected cloud for several continuous days. Therefore, we count multiple times (N_{miss}) of *true miss* to ensure the target lake really disappear. The value of N_{miss} is listed in Table 1.

6.1. Detect Lakes Appearing

We define a dark spot as lake when it appears for more than one day with similar area and location. For any spot, if we could find it twice within D_{valid} days, it is possibly a lake and we start to track it. This spot is a lake if it shows N_{show} times eventually, where the N_{show} appearances do not have to be in consecutive days. However, for this spot, if there are N_{miss} times of true misses before it appears N_{show} times, we conclude this spot is not a lake and ignore it. Values of D_{valid}, N_{show}, and N_{miss} depend on the geophysical properties of supra-glacial lakes. Our experimental values are listed in Table 1.

6.2. Track Lake Changes and Disappearances

After identifying a spot as lake, we track it until it disappears. In practice, we track each lake in the subsequent images until there are true misses for N_{miss} continuous days. To track lakes, we search the *corresponding spot(s)* in the *next image*. This way could track most lakes, which change slowly in days, and it also works in lakes with sudden change. Lake area change dramatically when drain or merge and it is hard to differentiate these sudden changes from noise. However, these sudden changes are infrequent and lake area changes slowly after that. Therefore, we could track two "slowly changing lakes" before and after the sudden change. If those two lake *map* with each other, they are one identical lake.

Assuming there is a lake L in day D, the tracking mechanism works as follows: Before reach N_{miss} continuous true misses,

1. if there are *corresponding spot(s)* of L in any of the following days $D + n$ (where $n > 0$), we say lake L still exists.
2. Otherwise, if we could find another pair of *corresponding spot(s)* (L' and L'') in the same location as L in the following days (i.e., $D + n$ where $n > 0$), we say L, L', and L'' are one identical lake.

6.3. Remove Non-lake Spots

In this section, we discuss how to improve accuracy by removing "persistant non-lake spots" according to lake properties. It is designed specifically for surpaglacial lakes and it might be unnecessary for readers how apply this tracking algorithm in another problem. Consider the life cycle of each supra-glacial lake. It starts from a shallow lake with medium reflectance (i.e., gray spots) and becomes a deeper lake with low reflectance (i.e., dark spots). In order to catch the whole life cycle of a lake, we include gray spots as possible lake candidates in Section 5. However, ice with high humidity would also appear as gray spots, while it would not turn into dark spots as lakes. The number of detected gray spots is small and most of them appear before the beginning of the melting season, usually in May in every year.

It is difficult to define gray and dark in images with large brightness variations. It is also not easy to find a reference point since snow reflectance changes humidity. However, ice in high elevation does not melt and its reflectance could be treated as the reference point in each image. Assume the reflectance of darkest pixel is 0% reflectance and brightest pixel is 100% reflectance. Based on the characteristics of our data, we look for 95% reflectance as dry snow reflectance in each image. We then define any spot with reflectance less than the half of the

dry snow reflectance as a dark spot. Otherwise, it is a gray spot. Any real lake composed of a set of temporal-varying spots should contain at lest one dark spot.

7. Evaluations

7.1. Experimental Setup

Moderate Resolution Imaging Spectroradiometer (MODIS) imagery (`http://modis.gsfc.nasa.gov/`) of 10 years' (Year 2000 to 2009) melting seasons in the west Greenland (Latitude: N 68.06 to N 70.43, Longitude: W 48.27 to W 51.92) were collected and used in our experiments. For each melting season, we choose May 1st as the start day (D_s) and October 1st as the end day (D_e). In order to conduct analysis in high resolution images (250 meter), we use only band 1 reflectance (optical wavelength: 620nm to 670 nm) from "MODIS Calibrated Radiances 5-Min L1B Swath 250m V005" (i.e., MOD02 swath data). Cloud masks and snow masks were extracted from the corresponding Snow Cover data: "MODIS/Terra Snow Cover 5-Min L2 Swath 500m V005" (i.e., MOD10-L2 swath data). All MODIS data were downloaded from the Warehouse Inventory Search Tool (WIST, `https://wist.echo.nasa.gov/~wist/api/imswelcome/`).

We compare all satellite images in the same coordinate system. All images were mapped into a 1000 row by 500 column grid using "MS2GT: The MODIS Swath-to-Grid Toolbox" (`http://nsidc.org/data/modis/ms2gt/`). The mapped images are centered on Latitude: 69.25, Longitude: -50 with resolution equal to 250 meter. A total of 9,684 MOD02 files and 9,040 MOD10-L2 files are collected and mapped. Among those images, 893 high-quality and less cloudy images were automatically selected by our system (Section 4) for lake tracking.

Table 1 and Table 2 summarize the key parameters and variables used in this work. Application parameters are listed in Table 1, such as maximum dimension of lakes (O_{SE}), correspond to the geophysical properties of our experimental data. Values of those parameters are specifically suggested by geo-scientist for detecting supra-glacial lakes in west-Greenland. Those values should be modified correspondingly when applying our method in different scenarios, such as detecting lakes above frozen soil in Alaska or temporal tracking of oil spill on the surface of sea. Key algorithm variables are listed in Table 1. Sensitivity analysis of the algorithm variables are presented in Section 7.5.

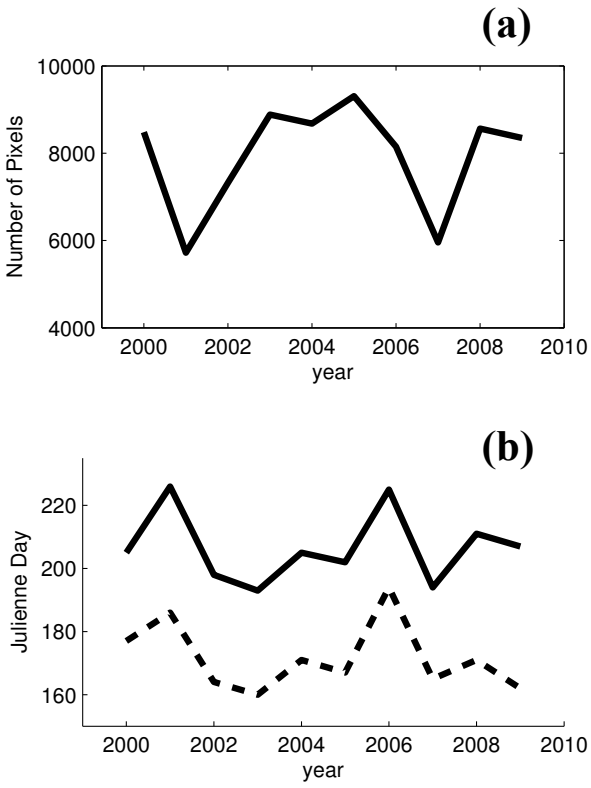

Figure 7: Statistics of tracked lakes over 10-year data (2000 to 2009). (a) total area of tracked lakes in each year. (b) dash line: median of lake starting day; solid line: median of lake closing day.

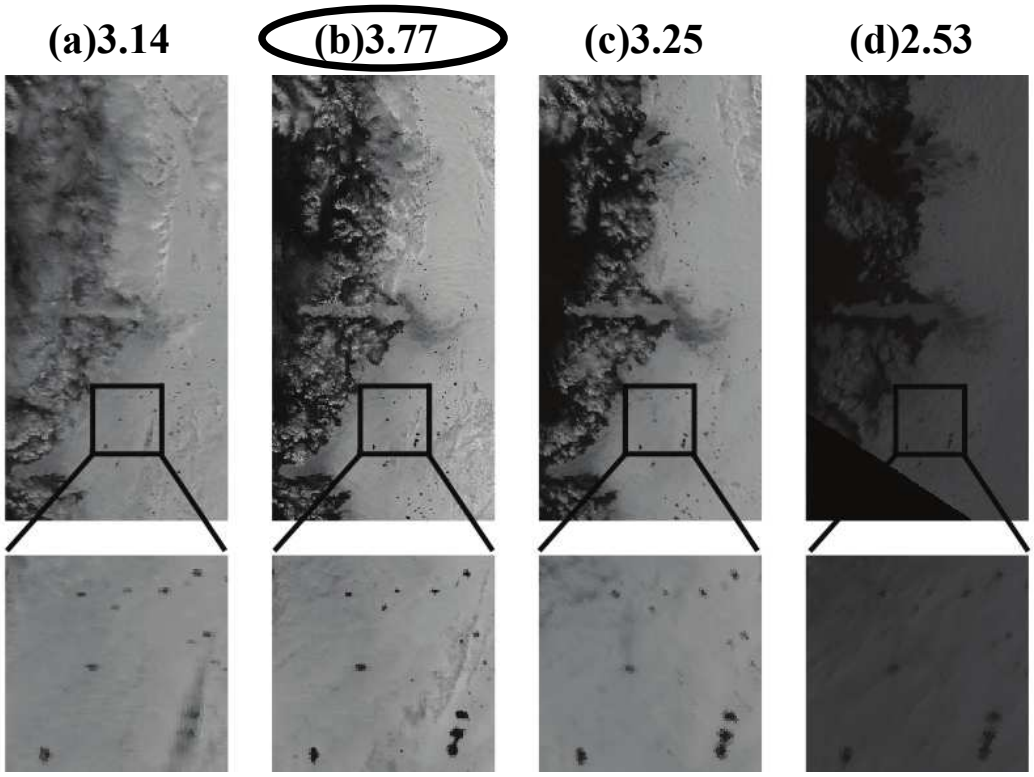

Figure 8: Select the best-quality image in each day: Those 4 images were all taken on July 4th, 2009. Quality coefficients (Equation 3) are listed above each image. Larger quality coefficient means better quality. (b) is the best-quality image and its quality coefficient is the highest among four. Eventually, our algorithm correctly picks (b) as the best image on that day.

Table 1: Application Parameters and Settings

Symbol	Description	Value
R	Number of rows in each mapped image	1000
C	Number of columns in each mapped image	500
D_s	Start day of analysis in each year	May 1st
D_e	End day of analysis each year	October 1st
E_{thr}	Minimum effective area of qualified images	$R \times C \times 0.3$
$\bar{I}_{thr}$	Minimum average intensity of qualified images	0.15
O_{SE}	Maximum dimension of lakes or size of outer structuring element	25
D_{valid}	time duration (in days) of first two appearing days of a legitimate lake	6
N_{show}	Number of appearances for a spot to be treated as a real lake	3
N_{miss}	Number of misses for a lake to be treated as a disappearing lake	5

7.2. Results Demonstration

We show one example of image selection in Figure 8. In each day, our algorithm picks the best-quality image based on the quality coefficient (Equation 3). Images in Figure 8 were all taken on July 4th, 2009. (b) is the best-quality image. (a) is more cloudy than (b). (c) is less cloudy than (b) but more blur. (d) is dark and blur. Quality coefficient (Q_k) for (a), (b), (c), and (d) are 3.14, 3.77, 3.25, and 2.53, respectively. Since larger quality coefficient means better quality, our system correctly selects (b) as the best image of that day.

Figure 6 demonstrates an instance of lake detection result. Each column is each day. From left to right is July 8th, 10th, 12th, and 13th in Year 2006. First row are the spot detection results in each image (Section 5) and brightest white pixels are detected spots. Second row are the original images. Third row are the results of temporal tracking (Section 6) with tracked lakes as brightest white pixels. The lakes reported by our system are the temporal tracking results (3rd row), which we refer to as the *tracked lakes*. We could see that, despite poor

Table 2: Algorithm Variables and Settings

Symbol	Description	Value
$X1$	Moving average window size of histogram = $N_{bin}/X1$*	40
$X2$	Moving average window size of 1_{st} derivative of histogram = $N_{bin}/X2$*	20
$X3$	1_{st} derivative of histogram of our contrast threshold	0.01
*N_{bin} = Number of bins in contrast histogram (Equation 6)		

image quality and cloud screen, temporal tracking can robustly report temporal-consistent lakes.

Figure 7 presents some statistical results of the tracked lakes. Figure 7-(a) is "total area of tracked lakes" vs. time (i.e. each year). Since lake area changes over time, we pick the maximum area (in pixels) of each lake and sum them up as the total lake area. Figure 7-(b) shows the temporal information of the tracked lakes. For each lake, starting day is the first day that our system found that lakes. Closing day is the last day the lake was seen before disappearing. Dash line denotes the median starting day of all lakes and solid line is the median of lake closing day.

On average, it took 477.3 seconds (less than 8 minutes) to detect and track lakes in each year. All the algorithms are implemented in MATLAB. The experiments were carried out on a workstation machine with 6GB memory and Intel(R) Xeon(R) CPU, 2.27GHz processor.

7.3. Evaluation of Tracked Lakes

We evaluate our temporal tracking results via comparing it with manual examination. Because manual examination is extremely time-consuming, we evaluated 4-year (2000, 2003, 2006, and 2009) instead of 10-year results and confine our examination region to 36% of our target region. Since our lake tracking results have marked out most lakes, we evaluate the results by identifying mistakes made by our algorithm. A lake is correctly detected and tracked if it (1) is indeed a lake, (2) has correct disappearing time, and (3) has correct lake area and shape.

The evaluation results are listed in Table 3 and Table 4. Assume the group of tracked lakes is *Result*. Group of manually identified lakes is true lakes or *Truth*. Evaluation is try to find how close of *Result* and *Truth*. In these tables, the meaning of the notations are listed as following:

- TP (true positive): Number of correctly tracked lakes, (i.e., *Result*∩*Truth*).
- FN (false negative): Number of true lakes which were not tracked by the algorithm.
- FP 1 (Type 1 false positives): Number of tracked lakes which are not ture lakes but other geographic features (e.g., rocks).
- FP 2 (Type 2 false positives): Number of tracked lakes which are indeed lakes but algorithm reported incorrect lake properties (e.g., wrong lake appearing day).
- Precision: It means the precision of *Result* (i.e., (*Result*∩*Truth*)/*Result*), or proportion of "correctly tracked lakes" to "all tracked lakes". In practice, it is "TP/(TP+FP1+FP2)".
- Recall: It shows how much percentage of *Truth* is recalled by *Result* (i.e.,(*Result*∩*Truth*)/*Truth*), or proportion of "correctly tracked lakes" to "all true lakes". In practice, it is "TP/(TP+FP1+FP2)".

Table 3: Lake Evaluation

Year	2000	2003	2006	2009
TP	331	312	282	269
FN	6	3	0	3
FP 1	4	3	7	5
FP 2	4	6	9	7
Precision	0.976	0.972	0.946	0.957
Recall	0.982	0.990	1.000	0.989

Table 4: Average Precision and Recall

Average Precision	0.963
Average Recall	0.990

7.4. Temporal Tracking vs. Spot Detection

Temporal tracking is the major reason why we could deal with cloudy and noisy images. In Figure 6, we could see that spot detection miss lakes in cloudy days (1^{st} row) while temporal tracking in the 3^{rd} row is much robust and insensitive to noise and cloud. Table 5 further demonstrates that temporal tracking actually removes noisy spots and recovers unseen lakes. Assume the "lakes detected by spot detection" is *Result* and "temporal tracked lakes is" *Truth*. In each image (1)L_T = Number of temporally tracked lakes, (2) L_S = Number of spots detected by spot detection algorithm, and (3) L_{TS} = Number of temporal tracked lakes which are correctly detected by spot detection algorithm. Recall and Precision of spot detection as follows and values are listed in Table 5.

- $Precision_{ST}$: L_{TS}/L_S
- $Recall_{ST}$: L_{TS}/L_T

Results in Table 5 tells us two things:

1. Temporal tracking removes 21.4% spots which are noise: "$Precision_{ST} = 0.786$" means, on average, 78.6% of detected spots are tracked in temporal tracking. In other words, 21.4% of detected spots are not consistent over time and very likely to be noise. Therefore, adding temporal tracking helps to remove those noises.
2. Temporal tracking recovers 43.5% lakes: "$Recall_{ST} = 0.565$" means, on average, only 56.5% of temporally tracked lakes were detected spot. The remaining 43.5% lakes, which are unseen becuase of cloud but still appear in the previous and following images. In other words, temporal tracking recovers those cloud-obscured lakes. Therefore, temporal tracking is the major reason why we could process cloudy images.

Table 5: Temporal Tracking vs. Spot Detection

Year	2000	2003	2006	2009	Average
$Precision_{ST}$	0.745	0.800	0.731	0.867	0.786
$Recall_{ST}$	0.517	0.578	0.506	0.660	0.565

7.5. Sensitivity Analysis of Algorithm Variables

We discuss how sensitive of system result to fluctuation of algorithm variables (Table 2). Application parameters (Table 1) should be defined by the domain expert and we do not discuss them here. The three variables (i.e., $X1$, $X2$, and $X3$) are all related to dynamic threshold searching in spot detection (Section 5.2). Briefly speaking, spots are composed of prominent pixels with contrast higher then a contrast threshold. We search a contrast threshold dynamically by the feature of histogram shape. Because histogram are very spiky, $X1$ and $X2$ are defined to smooth histogram and stablize result. On ther other hand, the mechanism of $X3$ is similar to "threshold", since threshold is "a larger-than-zero and smallest contrast where histogram 1^{st} derivative is equal to $X3$". More details could be found in Section 5.2. The following analysis shows that (1) result is insensitive to $X1$ and $X2$, and (2) result changes with $X3$ and users could generate preferable results by tuning $X3$ alone.

We define *average lake percentage* for evalutating sensitivity. *Lake percentage* is percentage of detected lake pixels in each image. In here, we use year 2009 data to evaluate sensitivity and *average lake percentage* is average of all *Lake percentage* in year 2009. Therefore, if we perturb one variable and cause large variation in *average lake percentage*, it means result is highly sensitive to this variable.

Let's examine the mechanism of $X1$ first. By moving average $X1$ is defined to smooth histogram and stablize result. The size of average window (i.e., $N_{bin}/X1$) is inversely proportional to $X1$. Therefore, larger $X1$ (i.e., smaller window) results in noisier curve but fit the histogram better (i.e., red solid line in Figure 9). On the other hand, smaller $X1$ (i.e., larger window) resulteds smoother curve but not fit the histogram well (i.e., red dashed line in Figure 9). We also could see that, compare to large $X1$, smoothed curve with small $X1$ reach "close-to-zero" value slower. Since threshold is the contrast where "close-to-zero" value is, small $X1$ results in larger threshold and less spot pixels. In other words, smaller $X1$ result in smaller *Average lake percentage*.

Figure 10-(a) shows that *Average lake percentage* (1) did decrease when $X1$ become smaller, and (2) is not sensitive to $X1$. We choose a wide range of $X1$ (i.e. 5 to 200) and *Average lake percentage* from 0.27% to 0.32%. The average is 0.3% and standard deviation is 0.014%. Therefore, the coefficient of variation, which is (standard deviation)/(mean), is only 0.047. In other words, result is not sensitive of fluctuation of $X1$.

Average lake percentage is also insensitive to $X2$ (Figure 10-(b)). Function of $X2$ is similar to that of $X1$ and *Average lake percentage* also reduces when $X2$ is

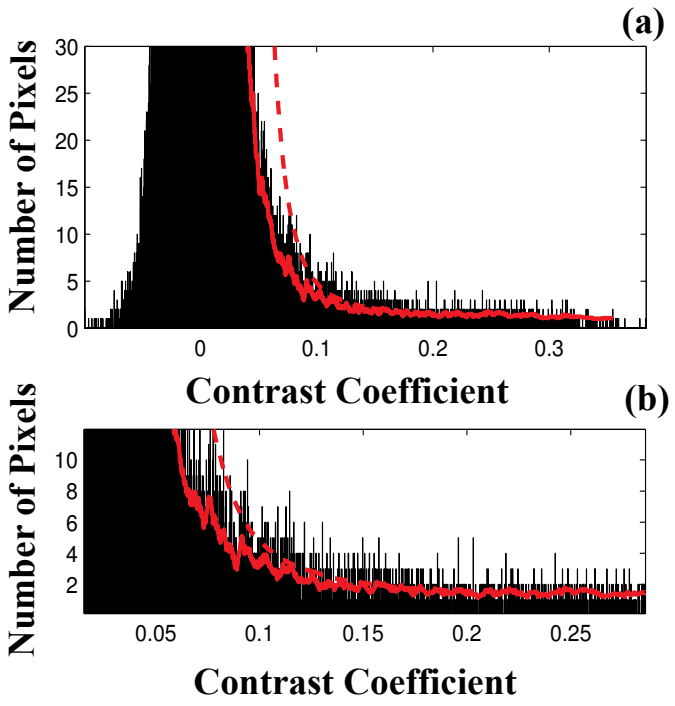

Figure 9: In both figures, y-axis is the number of pixels and and x-axis is the contrast coefficient. (a) Black histogram: one contrast histogram of the best image of day 168, year 2009. Solid red line: smoothed curve when $X1 = 200$. Dashed red line: smoothed curve when $X1 = 10$. (b) is the enlarged view of (a).

smaller. We could see in Figure 10-(b) that perturbing $X2$ from 5 to 100 changes the average lake percentage from 0.27% to 0.34%. The coefficient of variation is 0.07, which is still very small.

Scientiests could generate preferable results by tuning $X3$ alone. Compare with $X1$ and $X2$, *Average lake percentage* is much more sensitive to $X3$ and this is desirable. We could see in Figure 10(c) that a wide variation of $X3$ (i.e., 1e-4 to 1e-1) changes the average lake percentage from 0.19% to 0.82% and the coefficient of variation is 0.566. In other words, varying $X3$ could dramatically change the results. The mechanism of $X3$ is similar to "threshold", which control whether the gray pixels are included as spots or not (Section 5.2). Since result is insensitive to $X1$ and $X2$, scientiests could control end result by only varying $X3$.

8. Algorithm Summary and Related Works

Our interdisciplinary work, which geo-temporal track supra-glacial lakes on the Greenland ice sheet, draws upon research in a number of fields. In this section, we survey research works that are most related to ours.

8.1. Spot detection

Lake detection is similar to spot detection, since supra-glacial lakes appear as prominent spots above ice sheet (Figure 1). "Spot" is an essential visual feature and computer-aid spot detection algorithms are widely applied in many fields, such as electrophoresis in Pleiner et al. (1999), Astrophysics in Cruz et al. (2005), Agriculture in Mertens et al. (2005), Dermatology in Miyamoto et al. (2002), oil spill in Shua et al. (2010), medical imaging in Reiser et al. (2008), Duarte et al. (2010), etc. Based on the properties of supra-glacial lakes, we focus on methods which

- Are unsupervised methods which do not need trainning data,
- Could detect spots with "regular or irregular shapes" and "fixed or continuous intensities",
- Are insensitive to heterogeneous backgrounds, and
- Could automatically determine threshold while there is huge variation in image qualities.

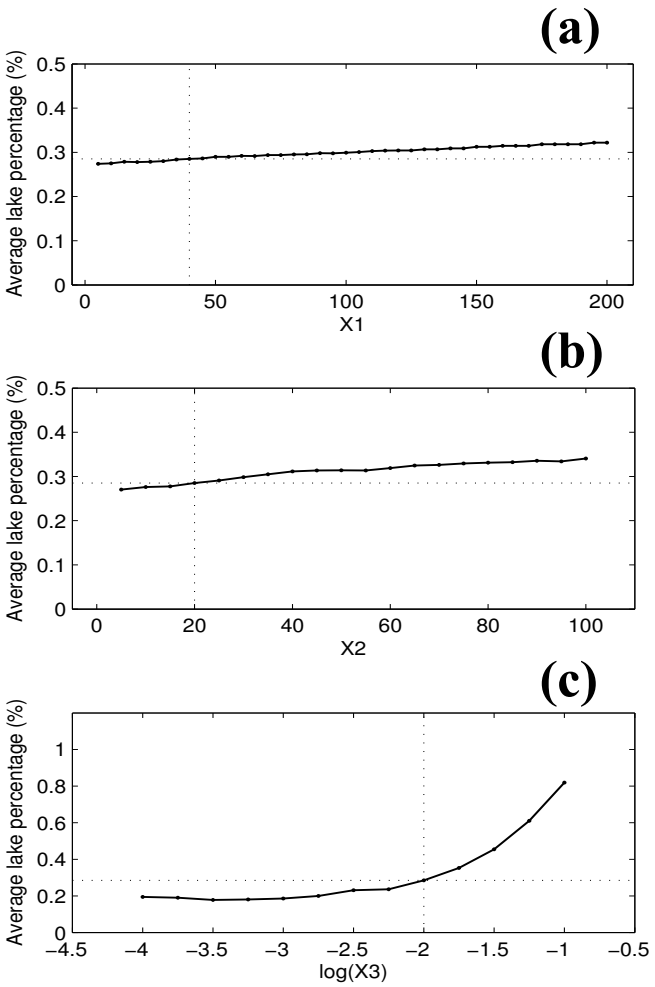

Figure 10: Sensitivity of results (average lake percentage) to algorithm variables. Dashed cross line marks the variable value we used in the lake result. Coefficient of variation (CV), which is (standard deviation)/(mean), is a dimensionless measurement of sensitivity. CV of each variable is (a)X1: 0.047, (b)X2: 0.07, and (c)X3: 0.566. Larger CV means the result is more sensitive to the variable. Therefore, (1) result is insensitive to $X1$ and $X2$, and (2) scientists could generate desirable results by only varying $X3$.

We developed a novel unsupervised spot detection algorithm which achieve all four requirements. Briefly speaking, it identified spots as high-contrast pixels. 'Ccontrast" is intensity difference (i.e., "reflectance difference" in our data) of each small region (i.e., "one pixel" in our data) and its neighboorhood. Plot contrast distribution as a histogram, we automatically search contrast threshold based on the feature of histogram shape. Pixels with contrast which are higher than threshold are spots. Our algorithm satisfies all aforementioned criteria and is efficient and accurate. For example, only a few number of convolution operations (1 to 2 convolutions in our experiments) are needed for each spot detection operation. Besides, shapes of detected spots are not distorted and we do not have to further correct lake contours.

We briefly review some related algorithms while none of them meet all aforementioned requirements.. Chu et al. (2009) Sundala et al. (2009) detected supraglaical lakes by classification methods, which are supervised methods and trainning data is necessary. However, applying supervised method in a large dataset with huge variation is time-consuming and resulted model might not be generalized enough. Conventional clustering methods, such as k-means or the well-known Otsu's method by Otsu (1979), are unsupervised but not suitable for our task. Because area of lakes is much smaller than that of ice sheet and clustering method ignore tiny cluster of lakes and divide ice sheet as two clusters instead. We also looked into spot detection algorithms (e.g., Smal et al. (2009); Ruusuvuori et al. (2010)), especially the ones which could extract spots from inhomogeneous background. Spot detection based on multi-scale wavelet product by Olivo-Marin (2002) caught our eyes first. It calculates local contrast by wavelet transform in different scales and flags pixels which wavelet product exceeding threshold in one or several scales. This method could extract spots from inhomogeneous background and search threshold dynamically. However, 2-D wavelet is not good in detecting irregular shape. And, for each spot detection task, many scaled wavelet convolution is necessary to cover all the lake dimension (i.e., from 1-pixel to 20-pixel wide in both X and Y direction). Since we need to apply spot detection many times (i.e., 6000 times in our experimental data), large number of convolution operations is time-consuming and make algrithm inefficient. Another possible choice is h-dome transformation by Vincent (1993), which could deal with irregular-shape spots and inhomogeneous background. We did not use it is because a good choice of "h" in h-dome transformation depends on the image properties. In other words, we still need to figure out a way to dynamically search h for each image. Besides, image reconstruction later on is still time-consuming. Since none of those exactly met our need, we choose to develop our own spot

detection algorithm (Section 5).

8.2. Spot Tracking

The biggest challenge for tracking time-varying area on Earth surface, such as supra-glacial lakes, is how to eliminate influence from cloud. The reason is that it is difficult to detect cloud above ice and cloud show unexpectedly, Roy et al. (2005) did a similar work and they tracked fire-affected areas using MODIS time series data. The basic idea is that reflectances sensed within a temporal window are used to build a model to predict the reflectance on a subsequent days. Built model is based on physical properties of remote sensing system and interested surface. Than the model is used for predicting reflectance in different viewing direction. If the difference between the predicted and observed reflectance is significant, a geographical change happens. To make this method work well, we need to have accurate cloud information most of time. However, this method would not work in our data since the best cloud distribution we could get contains lots mistakes.

The tracking method we present here is suitable for tracking spots with unexpected noise (e.g., cloud), such as remote sensing of Earth surface changes. The major concept is temporal consistency of spots. Specifically, for one spot present consistant through time (i.e., several frames), it is not a noise. Our method could also tolerate uncertainty of cloud information. The tolerance of unexpected cloud could be tuned by users according to different data properties. The given problem need to met the following criteria:

- Temporal resolution of images capture temporal variation of spots
- Noise (i.e., cloud) is not as persistant as target spots (i.e., lakes)
- Spots could merge, split, or change shape and area.

Since dimension of "spots" could define by users, this framework is suitalbe for lots remote sensing task, such as tracking pine beetle forest infection in Wulder et al. (2006), oil spill in Shua et al. (2010), or other fields like monitoring cell culture, etc. The detailed technique is described in Section 6.

9. Conclusions and Future Work

In this work, we have designed and developed a complete solution for the problem of geo-temporal tracking of supra-glacial lakes on the Greenland ice sheet.

We propose novel techniques to (1) accurately select the best-quality satellite image for each time interval (e.g., daily); (2) robustly detect possible lake candidates in each single image using automatic contrast thresholding, which performs well regardless of the diverse qualities of input images; and (3) leverage temporal consistency information to track real lakes over time and record desired information.

For lake candidate detection in single image, we purpose a novel unsupervised spot detection method. It could extract prominent dark spots with (1) irregular shapes, (2) continuous intensities, (3) heterogeneous backgrounds, by automatic thresholding. It only needs to execute a few number of convolution operations and the shape of detected spots is not distorted.

This framework could track irregular-shape / homogeneous object in cloudy images and is suitalbe for most remote sensing Earth surface change tasks. In our solution, after combining spot detection results with temporal registering of lakes, we could correctly identify and track lakes over time using partially cloudy images.

Our solution is highly efficient and achieves high accuracy. 96.3% tracked lakes are supra-glacial lakes (precision is 0.963 over 1.0), and 99.0% supra-glacial lakes could be found by our tracking algorithm (recall is 0.990 over 1.0). It took less than 8 minutes to track all the lakes in one year MODIS data. Detailed evaluation results, which is based on real satellite image over 10 years, demonstrate that the proposed solution can detect and track supra-glacial lakes under a variety of geographical and temporal conditions.

As our future work, we are in the process of obtaining more MODIS data and plan to conduct larger-scale studies that cover more geographical regions and temporal periods. We will also collaborate with domain experts to interpret the results we have and analyze the correlations between supra-glacier lake changes and glacial movements.

References

Bartholomew, I., Nienow, P., Mair, D., Hubbard, A., King, M. A., & Sole, A. (2010). Seasonal evolution of subglacial drainage and acceleration in a greenland outlet glacier. *Nature Geoscience*, *3*, 408–411.

Box, J. E., & Ski, K. (2007). Remote sounding of greenland supraglacial melt lakes: implications for subglacial hydraulics. *Journal of Glaciology*, *53*.

Chu, V., Smith, L., Rennermalm, A., Forster, R., Box, J., & Reehy, N. (2009).

Sediment plume response to surface melting and supraglacial lake drainages on the Greenland ice sheet. *Journal of Glaciology*, *55*, 1072–1082.

Cruz, M., Martínez-González, E., Vielva, P., & Cayón, L. (2005). Detection of a non-Gaussian spot in WMAP. *Monthly Notices of the Royal Astronomical Society*, *356*, 29–40.

Das, S. B., Joughin, I., Behn, M. D., Howat, I. M., King, M. A., Lizarralde, D., & Bhatia, M. P. (2008). Fracture propagation to the base of the greenland ice sheet during supraglacial lake drainage. *Science*, *320*, 778–781.

Dengler, J., Behrens, S., & Desaga, J. (1993). Segmentation of microcalcifications in mammograms. *IEEE Transactions on Medical Imaging*, *12*, 634–642.

Duarte, M., Alvarenga, A., Azevedo, C., Infantosi, A., Pereira, W., Bamidis, P., & Pallikarakis, N. (2010). Microcalcifications segmentation procedure based on morphological operators and histogram filtering. In *MEDICON 2010, IFMBE Proceedings 29* (pp. 355–358).

Joughin, I., Das, S. B., King, M. A., Smith, B. E., Howat, I. M., & Moon, T. (2008). Seasonal speedup along the western flank of the greenland ice sheet. *Science*, *320*, 781–783.

Mertens, K., Ketelaere, B. D., Kamers, B., Bamelis, F., Kemps, B., Verhoelst, E., Baerdemaeker, J. D., & Decuypere, E. (2005). Dirt detection on brown eggs by means of color computer vision. *Poultry Science*, *84*, 1653–1659.

Miyamoto, K., Takiwaki, H., Hillebrand, G., & Arase, S. (2002). Development of a digital imaging system for objective measurement of hyperpigmented spots on the face. *Skin Res Technol.*, *8*, 227–235.

Olivo-Marin, J. (2002). Extraction of spots in biological images using multiscale products. *Pattern Recognition*, *35*, 1989–1996.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*, 62–66.

Pleiner, K.-P., Hoffmann, F., Kriegel, K., Wenk, C., Wegner, S., Sahlström, A., Oswald, H., Alt, H., & Fleck, E. (1999). New algorithmic approaches to protein spot detection and pattern matching in two-dimensional electrophoresis gel databases. *ELECTROPHORESIS*, *20*, 755–765.

Reiser, I., Nishikawa, R., Edwards, A., Kopans, D., Schmidt, R., Papaioannou, J., & Moore, R. (2008). Automated detection of microcalcification clusters for digital breast tomosynthesis using projection data only: a preliminary study. *Med Phys.*, *25*, 1486–1493.

Roy, D., Jin, Y., Lewis, P., & Justice, C. (2005). Prototyping a global algorithm for systematic fire-affected area mapping using MODIS time series data. *Remote Sensing of Environment*, *97*, 137–162.

Ruusuvuori, P., Aijo, T., Chowdhury, S., Garmendia-Torres, C., Selinummi, J., Birbaumer, M., Dudley, A., Pelkmans, L., & Yli-Harja, O. (2010). Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images. *BMC bioinformatics*, *11*, 248.

Shua, Y., Li, J., Yousifa, H., & Gomesa, G. (2010). Dark-spot detection from sar intensity imagery with spatial density thresholding for oil-spill monitoring. *Remote Sensing of Environment*, *114*, 2026–2035.

Smal, I., Loog, M., Niessen, W., & Meijering, E. (2009). Quantitative comparison of spot detection methods in live-cell fluorescence microscopy imaging. In *Proceedings of the Sixth IEEE international conference on Symposium on Biomedical Imaging: From Nano to Macro* (pp. 1178–1181). IEEE Press.

Sundala, A., Shepherda, A., Nienowa, P., Hannab, E., Palmera, S., & Huybrechtsc, P. (2009). Evolution of supra-glacial lakes across the greenland ice sheet. *Remote Sensing of Environment*, *113*, 2164–2171.

Vincent, L. (1993). Morphological grayscale reconstruction in image analysis: Applications and effective algorithms,". *IEEE Transactions on Image Processing*, *2*, 176–201.

Wulder, M., Dymond, C., White, J., Leckie, D., & Carroll, A. (2006). Surveying mountain pine beetle damage of forests: A review of remote sensing opportunities. *Forest Ecology and Management*, *221*, 27–41.

Zwally, H., Abdalati, W., Herring, T., Larson, K., Saba, J., & Steffen, K. (2002). Surface melt-induced acceleration of Greenland ice-sheet flow. *Science*, *297*, 218.