Automated Tracking of 2D and 3D Ice Radar Imagery Using Viterbi and TRW-S

Victor Berger, Mingze Xu, Mohanad Al-Ibadi, Shane Chu, David Crandall, John Paden, Geoffrey Fox

***Abstract*—Radar depth sounding and imaging systems produce two-dimensional and three-dimensional imaging of the subterranean structure of polar ice sheets. Information such as ice thickness and surface elevation is extracted from this data and applied to research problems in ice flow modeling and ice mass balance calculations. Due to the large amount of data collected, we seek to automate the ice-bottom layer tracking and also allow for efficient manual corrections when errors occur in the automated tracking. In this work, we present improvements made to previous implementations of the Viterbi and TRW-S algorithms for ice-bottom extraction in 2D and 3D radar imagery. Along with an explanation of our modifications, we demonstrate the results obtained by our modified implementation of the two algorithms and by previously proposed solutions to this problem, when compared to the available manually-corrected ground-truth data. Furthermore, we perform a self-assessment of tracking results by analyzing differences in the estimated ice-bottom for surveyed locations where flight paths have crossed and thus two separate measurements have been made at the same location.**

***Index Terms*—Feature extraction, glaciology, ice thickness, ice tracking, radar tomography**

I. Introduction

The Center for Remote Sensing of Ice Sheets (CReSIS), based at the University of Kansas, designs and develops radar instrumentation that allows for wide-coverage remote sounding and imaging of the ice sheets, snow, and sea ice in polar regions.

The data acquired by these sensors provide information about the basal topography of the ice structures of the surveyed region, from which measurements such as ice thickness can be derived. Analysis of this polar topography data helps determine the contribution of ice caps to the present sea level using the surface mass balance and discharge method, and can be factored into ice-flow modeling studies to predict their future impact on the sea level [1].

The raw data collected by systems such as the Multichannel Coherent Radar Depth Sounder [2] pass through a processing pipeline including pulse compression, synthetic aperture radar processing, and array processing steps to resolve the data in the range, along-track, and cross-track dimension, respectively [3]. Data products from this processing include 2D SAR images and 3D SAR tomography, both of which display the structure of the subterranean ice topography.

In both of these data products, the most relevant features are the ice-surface and ice-bottom layers. The former is the interface between the air and the ice; the latter is the interface between the ice and the bedrock or liquid water underneath. The location of these layers in each echogram is used in the calculation of the ice thickness of the surveyed area, and thus some form of layer tracking is required.

In a typical deployment of the CReSIS depth sounding systems, thousands of kilometers of terrain are covered per day. Due to the large amount of data collected, accurate manual tracking of the 2D echograms is a slow and time consuming process, and effectively impossible in the case of 3D imagery where hundreds of thousands of images are generated if we view the 3D imagery as a stack of 2D images. This has driven the development of automated ice layer tracking systems.

This problem has received attention from researchers such as Gifford *et al.* [4], who proposed edge-based and active-contour-based iterative methods of tracking the interfaces. Similarly, a solution utilizing a level-set technique was suggested by Rahnemoonfar *et al.* [5]. Another approach was proposed by Crandall *et al.* [6], which poses this tracking as an inference problem, solved using the Viterbi algorithm [7] on a probabilistic graphical model which combines several known constraints of the polar ice layers. An improved solution using a similar model was suggested by Lee *et al.* [8], which employed a Markov Chain Monte Carlo (MCMC) technique to solve the inference problem and also allowed the ice-surface and ice-bottom to be simultaneously solved. Another solution was proposed, this time specifically for 3D images, by Xu *et al.* [9], using a sequential tree-reweighted message passing (TRW-S) [10] technique. This problem was later revisited by Xu *et al.* [11] using deep convolutional and recurrent neural networks.

We present an adaptation to the aforementioned Viterbi [6] and TRW-S [9] solutions that adjusts the cost function to better match the dataset and include additional evidence. We apply the adapted algorithms to CReSIS data, and assess the results obtained by these techniques in terms of tracking accuracy and speed. We determine average tracking errors by comparing to the available manually-tracked ground-truth data, and compare this with previous solutions. We also check the self-consistency of the 3D algorithms by comparing results where flightlines cross and two independent measurements have been made at the same location.

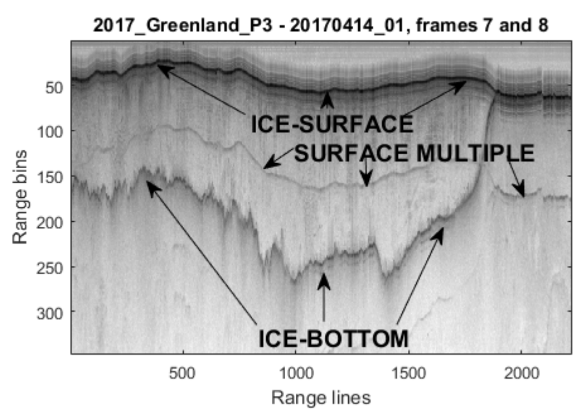

Fig. 1. Example of 2D echogram. Notice that the ice-surface and ice-bottom merge on the right when no ice is present.

This work was supported by NSF (1443054). Victor Berger, Mohanad Al-Ibadi, and John Paden also acknowledge support from NASA (NNX16AH54G). ArcticDEM was provided by the Polar Geospatial Center under NSF OPP awards 1043681, 1559691 and 1542736.

V. Berger, S. Chu, M. Al-Ibadi, and J. Paden are with the Center for Remote Sensing of Ice Sheets, University of Kansas, 66045 (e-mail: paden@ku.edu).

M. Xu, D. Crandall, and G. Fox are with the University of Indiana, Bloomington, IN 47405.

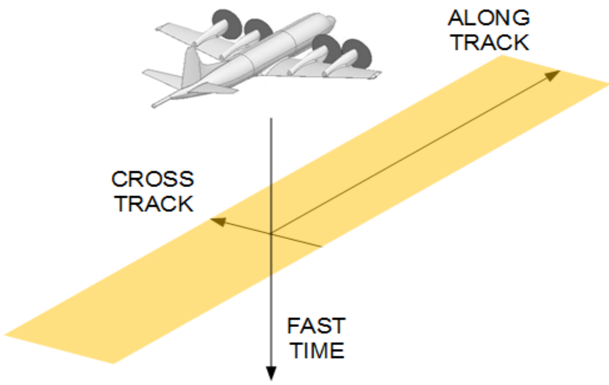

Fig. 2. Illustration depicting the axes of the 2D echograms relative to the radar platform. The cross-track offset is assumed to be zero.

The organization of this paper is as follows: in Section II, we describe the 2D and 3D formats of data on which layer tracking is performed, along with an overview of the tracking framework presented. In Section III, we define our modifications to the original cost functions, and in Section IV we present the algorithms used and our modifications to their implementation. In Section V, we present and analyze the results obtained, as well as a self-assessment of the accuracy of these results.

II. Background

A. Two-dimensional radar imagery

In a 2D image or echogram (e.g. see Fig. 1) the subterranean ice structures are displayed along the flight profile. The horizontal axis represents the along-track dimension, where each column is a range-line. The vertical axis corresponds to the fast-time dimension, where each row is a range bin. The vertical dimension is directly related to the depth of the subterranean ice structure. The pixel intensity is proportional to the radar scattering intensity with darker representing a stronger scattering signal. Fig. 2 illustrates the image axes with respect to the aircraft.

In Fig. 1, the very dark, continuous line near the top is the ice-surface, and the dark erratic line near the middle is the ice-bottom. Notice that these two interfaces merge around range-line 1,900 indicating the beginning of a region with no ice. Lastly, notice the layer under the ice-surface that follows its shape: this is the first surface multiple, and is always located at twice the time delay as the surface. This a signal processing artifact that may confuse the tracker and create a false positive, complicating the layer tracking process. A simple

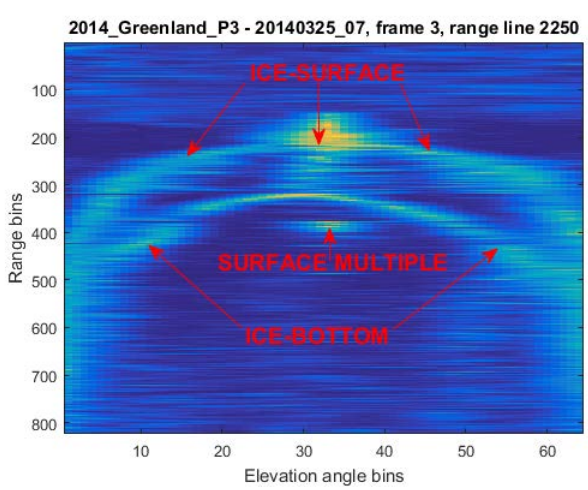

Fig. 3. Example of a 3D image slice.

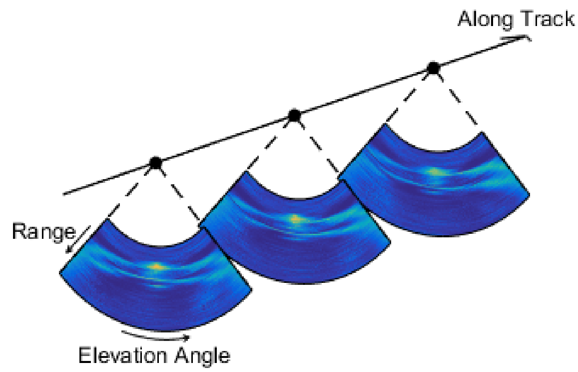

Fig. 4. A sequence of cross-track slices generates a 3D image of the surveyed terrain.

solution for mitigating the effects of this undesirable feature is described in Section III.

B. Three-dimensional radar imagery

The 3D images represent a sequence of cross-track images (or "slices") of the terrain. In each slice, the horizontal axis displays the cross-track elevation angles discretized into direction-of-arrival bins, and the vertical axis depicts the fast-time dimension in the same manner as a two-dimensional echogram where each row corresponds to a range-bin. Fig. 3 shows an example image slice, and Fig. 4 shows how the slices fit together relative to the radar platform coordinate system. The MUltiple SIgnal Classification (MUSIC) algorithm is used to generate the images [12]. The color of each pixel represents the MUSIC cepstrum which is *loosely* related to how likely a scatterer is present. In the figure, yellow indicates a larger cepstrum value which is associated with increased likelihood and blue indicates a lower cepstrum value.

C. Layer tracking framework for 2D and 3D imagery

While both formats of radar imagery display the subterranean ice sheet structures of the surveyed area, an important difference between the two is that there exists a strong correlation in two dimensions for the ice-bottom in the 3D image; that is in both elevation angle and along-track dimensions since the bottom layer tends to vary smoothly. On the other hand, no "third dimension" is available for the two-dimensional echograms. The 2D image is a subset of the 3D image and does not have an elevation angle dimension; it corresponds to just the nadir elevation angle bin (bin 33 in Fig. 3) from each 3D slice.

As such, different algorithms produce optimal layer tracking results for each image type. In the 3D case for example, the best algorithms exploit the layer correlation in two dimensions. In the 2D case, the reduced dimensionality of the data allows an exact solution to the minimization to be found (although less information is supplied to the minimization and so the result is generally worse than the 3D case [9]).

For all algorithms, we make the assumption that the ice-surface layer is known *a priori*, since there generally exist accurate ice-surface estimates (such as ArcticDEM and Bedmap2 [13]) based on satellite imagery for surveyed locations. The location of the ice-surface is an input to the ice-bottom tracker and is used to define portions of the cost function explained below.

Furthermore, we constrain the ice-bottom layer to be single-valued everywhere with respect to the elevation angle and along-track dimension, meaning that only one row can correctly label each column of the image matrix. In a physical sense, this is the same as assuming that the subterranean ice structure surveyed contains no overhangs or cave-like features from the perspective of the radar.

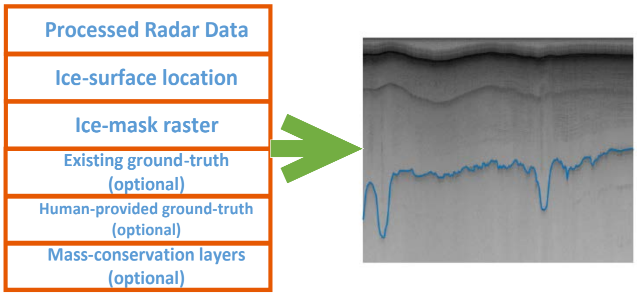

Fig. 5. Diagram illustrating main inputs of the automated layer-tracking systems. The location of the ice-bottom layer, shown here as a continuous blue line overlaid on an echogram, is the output for both 2D and 3D trackers.

III. A Graphical Model for Layer-Tracking

In previously published works, [6] and [9] pose the echogram layer-tracking problem as an inference problem on a statistical graphical model. For both 2D and 3D imagery, the authors assign a unary cost function ψ_U to every pixel, which represents the cost for the ice-bottom layer to pass through that pixel, and assign a column-to-column layer transition cost in a binary cost function ψ_B.

Based on these pixel and transition costs, a Markov Random Field (MRF) framework is formulated and inference is performed in order to find the lowest total cost solution. For both algorithms, the hidden states of this model are the rows that correctly label the ice-bottom layer in each column of the image matrix. While the method of performing inference on the MRF and ultimately detecting the lowest-cost (highest probability) solution differs between the two algorithms, in both cases the unary and binary costs are determined in a very similar manner.

The tracking output for the 2D imagery is in the form of an Nx-dimensional vector, where Nx is the number of range-lines in the input image. In the case of 3D imagery, the output is an Nb-by-Nx matrix, where Nb represents the number of direction-of-arrival bins in the three-dimensional image matrix.

Here, we present an expanded explanation of this layer tracking process, as well as modifications that have improved the accuracy of the results.

A. *Unary cost function*

The unary cost function, ψ_U, which assigns a cost to each individual pixel in the input image, has five terms which are explained below. Fig. 5 provides a summary of these inputs to the layer tracking software.

The first term ψ_{SURF} enforces the constraint that all points of the ice-bottom layer must be at or below the ice-surface. For the two layers to be at the same depth, it must be the case that no ice is present at that position, effectively generating an ice thickness measurement equal to zero.

Since "ice mask" datasets, which show where ice is present, are available for most regions surveyed by CReSIS (e.g. Randolph Glacier Inventory [17]) the algorithms will 1) merge the ice-bottom to the ice-surface where there is no ice ($MASK = 0$) and 2) force the ice-bottom to lie in a certain range relative to the surface if close to the ice margin (i.e. near the transition between no-ice and ice), and 3) have no effect is more than a certain distance away from the ice margin. Also, a value of positive infinity is assigned to the ψ_{SURF} term for all pixels located above the ice-surface, which guarantees that the bottom will always be below the surface.

Because a relatively smooth transition is expected between icy and non-icy regions of the terrain, the ice-bottom is restricted to a range of values near the ice margin. This is done by shrinking the icy regions by 2 pixels and then filtering this shrunk ice mask with a boxcar window equal to $2.56 \times [1\ 1\ 1\ 1\ 1]$ in the elevation angle dimension and then in the along-track dimension. Values above 90 are then set to infinity. For example, a binary ice mask sequence [0 0 1 1 1] would become a filtered ice mask sequence [0 0 32 65 ∞]. This cost function term is then given by:

$$\psi_{SURF}(s) = \begin{cases} \infty, & s > SURF \text{ or } (s \neq SURF \text{ and } MASK = 0) \\ & 0, SURF > \ s > SURF - MASK \end{cases}$$

where s represents the row index of the pixel of interest, $SURF$ is the row index of the ice-surface layer, and $MASK$ is the filtered ice mask.

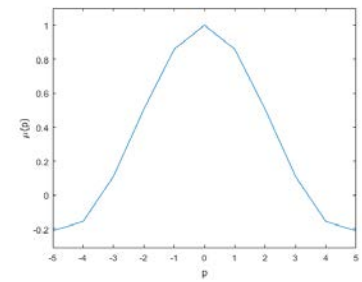

Fig. 6. Values of the correlation function $\mu(p)$ used in the unary cost calculation.

We modified the previously proposed template term in order to better use the dynamic range of the imagery. The ψ_{TEMP} term used in [6] and [9] measures the squared difference in the image pixel intensity relative to a template of an ideal layer return. The template was found through an automated training sequence using the *a priori* surface information. Although it is data dependent, the template invariably has a peak in the center with decreasing values towards the edges of the template. Because the ψ_{TEMP} term measured the squared distance to the template, a peak response in the imagery with the same intensity produces the lowest cost. The issue with using the squared distance to the template is that the peak intensity from the ice-bottom layer varies with larger intensities generally indicating a better measurement (since these correspond to greater signal to noise ratios). To better handle peak intensity variability, we now use a correlation operation that multiplies the template with the image:

$$\psi_{SINC}(s) = -\sum_{p \in T} I(s+p)\mu(p)$$

where $p \in T = \{-5, -4, -3, \ldots, 5\}$ refers to the pixel index of the correlation function, and $\mu(p)$ is the correlation function which is now fixed to $sinc\,(5p\,/\,1.5)$ which for ± 5 pixels approximately corresponds to the midpoint in the first minimum on either side of the $sinc$ function peak at $p = 0$ as shown in Fig. 6.

The term ψ_{GT}, shown in Fig. 7, of the unary cost function encourages the ice-bottom layer to be drawn towards ground-truth points, if they exist. To account for potential small inaccuracies of the ground-truth, the algorithm is not forced to return an answer in which the ice-bottom is found to exactly match the location of that point, but

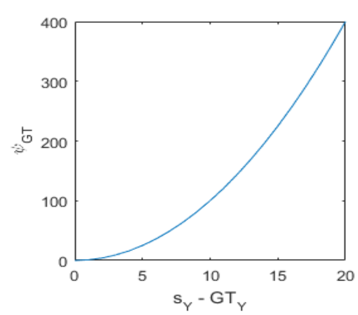

Fig. 7. The unary cost of every pixel in the input image is increased according to the squared vertical distance to ground-truth points in the same column of the image matrix, if they exist.

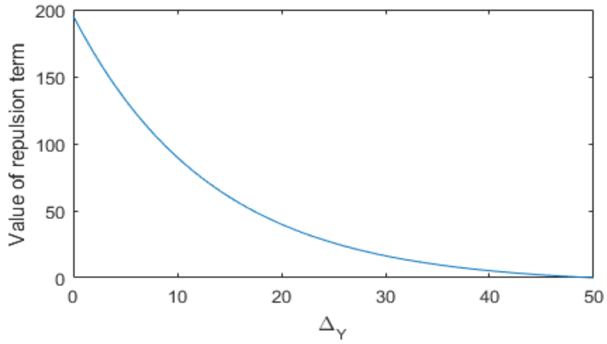

Fig. 8. Shifted exponential decay of the ψ_REP term as a function of the vertical distance Δ_Y between the pixel of interest and the ice-surface layer. The selected parameter values are α_MSD=50, α_MC=200, and λ=0.075.

encouraged to approximately match it by a cost term that is positively associated with the squared vertical distance to the ground-truth point, as in

$$\psi_{GT}(s) = (s - GT)^2$$

where s_Y represents the row index of the pixel of interest and GT is the row index of the ground-truth point.

Ground-truth can be manually added by a human operator. For 2D imagery, this is not done before the automated tracking is run. However, ground-truth points are automatically acquired by intersecting the flight path of interest with flight paths of previous surveys of that geographical region. Frequently, a given location will have been imaged and labeled before, and the point in which the new flightline crosses the old will already have ice-bottom depth information associated with it, which can then be used to help the tracker. These flightline intersections are commonly known as *crossovers*, and can also be used in determining the error associated with layer tracking results.

For the 3D imagery, ground-truth points are taken from the 2D tracking process by using the ice-bottom layer from the 2D imagery in the nadir elevation angle bin of the 3D imagery. Although this is not strictly required by the 3D algorithm, in all the results presented in this work, the nadir tracked bin from 2D imagery is used as ground-truth to the 3D imagery and we did not evaluate the performance without this ground-truth added in for 3D imagery.

Another potential sources of ground-truth are *a priori* estimates of the ice-bottom that can be used as weak evidence by the tracking algorithms via an additional unary cost term ψ_{EXTRA}. These are incorporated in the same way as manual or crossover ground-truth points, but with a lower weighting. These estimates can be obtained, for example, from existing ice thickness models based on ice flow dynamics and mass conservation [14].

Because the ice-surface return usually generates a strong and consistent region of high intensity pixels in the imagery, as is the case in Fig. 2, an ice-surface repulsion term ψ_{REP} is added to the unary cost function to prevent the tracker from incorrectly labeling the ice-surface as the ice-bottom. This is done by raising the unary cost of pixels that are within a certain maximum sensory distance (α_{MSD}) from the ice-surface. An ice thickness close to zero would cause a large increase in cost, defined by a maximum cost (α_{MC}) parameter, which would prevent the tracker from selecting it as the true label for the ice-bottom. To ensure a smooth cost increase as a function of proximity to the ice-surface, a shifted exponential decay function was chosen, as can be seen in Fig. 8. This term is calculated as

$$\psi_{REP}(s) = \begin{cases} 0, & \Delta_Y > \alpha_{MSD} \\ \alpha_{MC} * e^{-\lambda * \Delta_Y} - \alpha_{MC} * e^{-\lambda * \alpha_{MSD}}, & \text{otherwise} \end{cases}$$

where $\Delta_Y = SURF - s$ is the vertical pixel distance between the ice-surface layer and the pixel of interest s, and λ is the exponential decay constant.

The final unary cost of each pixel is calculated by a weighted summation of the aforementioned partial cost terms, as in

$$\psi_U = \psi_{SURF} + \psi_{SINC} + w_{GT}\,\psi_{GT} + \psi_{EXTRA} + w_{REP}\psi_{REP}$$

where the w variables are the weights assigned to each individual unary cost term and are discovered via hyperparameter optimization.

B. *Binary cost function*

The binary cost function assigns a cost ψ_B to every valid column-to-column transition in the image matrix. This function is tuned to enforce a smoothness constraint on the model, increasing the likelihood that transitions which generate smoother layers will be selected by assigning to these a lower cost. A smooth interface is generally a reasonable assumption for the bottom of the ice sheet.

In previous efforts, the implementation of this binary cost term was set to prioritize flat surfaces in the coordinate systems of the 2D and 3D imagery. However, since the 3D imagery are in a cylindrical coordinate system native to the radar sounding processing, this “flat” surface did not represent a flat surface in Cartesian space. A flat surface in Cartesian coordinates curves downward towards the edges of the 3D imagery (see Fig. 3 for an example of this effect). Also, if the aircraft altitude changes, both the ice-surface and ice-bottom will change together with altitude.

For this reason, in both 2D and 3D datasets the smoothing term has now been modified to set the lowest transition costs to rows that follow the range-slope of the ice-surface. In the 3D imagery scenario, although this is still not a flat surface in the Cartesian coordinate system, this is a flatter and more realistic approximation of the expected shape of the ice-bottom and computationally simpler than calculating the shape of a flat ice-bottom in Cartesian space which must account for ice refraction from a non-flat ice-surface layer. The new binary cost function term is given by:

$$\psi_B = w_B * (Source - Dest - Offset)^2$$

where $Source$ and $Dest$ are the row index of the source and destination columns respectively, and $Offset$ is the range-slope of the ice-surface between the column of interest and the previous column. A scaling factor w_B is used to define the weight of this smoothing term.

C. *Data pre-processing*

The 2D image intensity exhibits a strong dependence on depth in ice due to the ice loss and spherical spreading loss. We apply a simple detrending routine that normalizes the mean intensity of each row. This helps the tracker in areas where the bed echo is weak.

Without normalization, clutter near the ice-surface is often so strong that the ice-bottom layer tracker may jump up to this signal despite the layer smoothness and surface-repulsion constraints enforced by the unary and binary cost functions of the tracking algorithms. The previous proposed solutions of [6, 9] thresholded the image data to prevent the layer tracker from incorrectly tracking ice-surface points as the ice-bottom. The problem with thresholding is the loss of signal information associated with it. With the implementation of the detrending routine, data thresholding is no longer necessary.

While the 2D images are estimates of scatterer intensity, the 3D images are not. A similar detrending procedure is not as crucial in the 3D imagery because the MUSIC algorithm’s cepstrum produces a muted dynamic range.

As previously mentioned, an undesirable feature present in both the 2D and 3D imagery is the surface multiple, which is caused by a ringing of the radar signal between the ice-surface and the aircraft. To mitigate the effect of the surface multiple as a false positive to the algorithms, we have employed a simple method of decreasing the pixel intensity around the areas of the input image in which the surface multiple is located. It is possible to estimate the location of this feature by doubling the two-way travel time of the ice-surface.

Additionally, previous tracking efforts divided flight data into small frames for processing. These data frames are contiguous and this sometimes resulted in lower quality results near the edges of the data frames than is possible by processing the entire flight. For this reason, the two-dimensional data passed in to the Viterbi algorithm has been modified so that entire flights are processed at once. This also increases the probability that the layer data being processed will include ground truth from crossing lines, although this tends to have a relatively local effect on improving performance.

D. Parameter optimization

The weights and parameters in the cost function affect the accuracy of the automated tracker. Therefore, the optimal values of these parameters, such as the maximum sensory distance α_{MSD} and maximum cost α_{MC} of the ψ_{REP} term described above, need to be carefully tuned for the best possible performance of the algorithms.

For this purpose, multi-stage (multi-resolution) grid-search and random-search [15] parameter optimization techniques were used, supervised by the performance metric of ice-bottom layer mean error (measured in absolute pixel distance) when compared to a manually-tracked training set. Random search is a recent hyper-parameter global optimization technique that has been shown to outperform exhaustive grid-search methods in terms of accuracy and computational cost, particularly in large parameter spaces where not all variables have equal impact on the final error measurement and therefore are not equally important to tune.

Due to the differences in image structure between the 2D and 3D datasets, a distinct combination of optimal parameters was found for each case. Optimization was only performed using the Viterbi algorithm for the 2D imagery and TRW-S for the 3D imagery. The w_{GT} and w_{REP} weighting variables of the unary cost function were tuned. A scaling factor w_B for the binary cost function was also found. The optimal results found for these parameters is shown in Table 1.

TABLE I
PARAMETERS USED IN COST FUNCTION CALCULATIONS

	w_{GT}	w_{REP}	w_B
Viterbi	10	150	55
TRW-S	11	24	33

IV. ALGORITHMS APPLIED TO LAYER-TRACKING

Once both unary and binary costs have been assigned in the manner described above, we apply the Viterbi algorithm to the 2D and 3D imagery and the TRW-S algorithm to the 3D imagery to ultimately discover the lowest-cost label of the ice-bottom, which is taken to be the final result.

A. 2D imagery and the Viterbi algorithm

For the 2D imagery, we follow the solution proposed by Crandall *et al.* [6] of formulating a Hidden Markov Model (HMM) framework by splitting the MRF model into a set of non-loopy graphs, and then use the Viterbi algorithm [7, 16] to perform exact inference on each of these graphs in sequence.

Viterbi is an efficient dynamic programming method of finding the highest-probability sequence of hidden states in a finite-state discrete-time Markov process. This algorithm is guaranteed to return the global maximum likelihood path (the "Viterbi path") of an HMM.

In this framework the observed variables are each pixel of the input image data, and the hidden variables consist of the correct row labels for each column, as well as pairwise probability functions between neighboring hidden variables.

B. 3D imagery and the TRW-S algorithm

In order to take advantage of the strong correlation between consecutive slices of three-dimensional imagery, Xu *et al.* [9] proposes the use of a sequential tree-reweighted message passing (TRW-S) technique [10], in which cost information is passed both intra- and inter-slice. Because of the inter-slice message passing capability, this method is capable of preventing discontinuities in both along-track and elevation angle dimensions during the layer reconstruction.

Similar to the 2D solution, an energy minimization framework on a first-order MRF is formulated. While the intra-slice message passing procedure performs in similar fashion to the Viterbi algorithm by propagating evidence to its neighboring pixels to the left and right (direction-of-arrival dimension), the inter-slice message passing propagates ice-bottom layer evidence between consecutive slices of 3D imagery (along-track dimension).

The implementation of this algorithm has been changed from [9] so that the message passing along the direction-of-arrival dimension is now always performed outward from nadir, rather than switching from left-to-right and right-to-left message propagation on each iteration of the algorithm. The issue with the previous solution was that a strong preference was given to the cost messages originating from the extreme directions-of-arrival bins on either side, where the signal quality is usually the worst. Since we have ground-truth data at nadir (usually from having tracked the corresponding 2D dataset) and the signal quality is often best at nadir, the preferential message passing direction was changed to be always outward from nadir, in such way that the nadir column asserts the greatest influence on the final result.

After iterative belief propagation is performed, the set of labels that minimize the total (unary and binary) costs is selected as the answer.

Unlike the Viterbi algorithm, the TRW-S algorithm on an MRF is not guaranteed to converge to a global optimum. However, based on trial and error similar to [9], we found that 50 iterations usually produces satisfactory results. More systematic testing in the future may suggest convergence criteria rather than a fixed number of iterations.

C. 3D imagery and the Viterbi algorithm

The layer tracking solution using the Viterbi algorithm can also be applied to 3D imagery with no additional adaptations. This is accomplished by passing in individual slices of the 3D imagery to the algorithm. This input format differs from that of the TRW-S algorithm, to which three-dimensional matrices can be passed in.

In order to force propagation of layer evidence through the range-line dimension, the tracking result of a given slice may be passed in as ground-truth to the next slice in the 3D data frame, but we do not explore this possibility in this work.

However, as expected, when applied to three-dimensional imagery the Viterbi algorithm is outperformed in accuracy by the TRW-S algorithm and is more likely to generate discontinuities in results, particularly along range-lines due to the absence of message-passing in that dimension.

V. RESULTS AND DISCUSSION

A. 2D imagery

We tested our modified Viterbi routine on 2D data from the 2009 NASA Operation IceBridge Antarctica campaign, the same dataset used by the authors of [5] and [8]. The algorithm received no manual aid of any kind, and the only ground-truth points provided were the aforementioned automatically-acquired crossovers. We did not re-run the other results for the 2D imagery; rather, these are the results

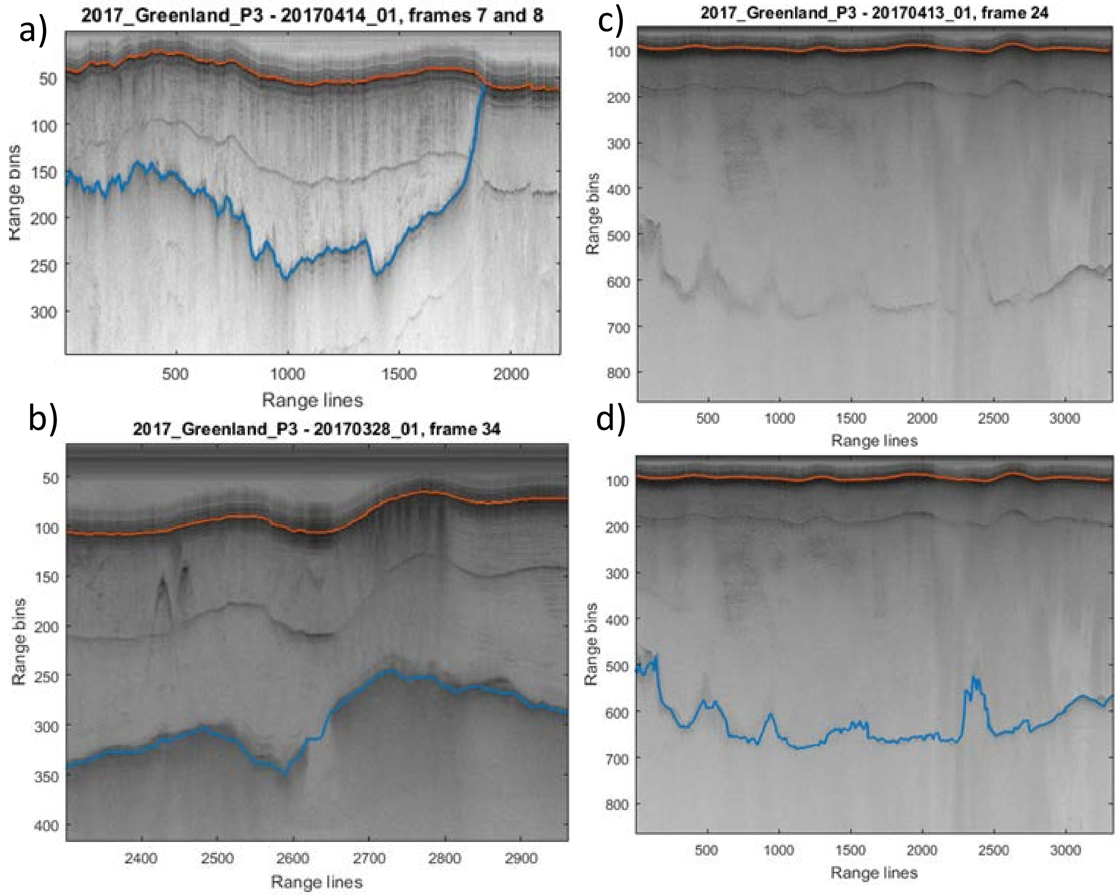

Fig. 9. a) Example of labeled 2D echogram displaying the known ice-surface layer and the ice-bottom layer tracked by our implementation of the Viterbi algorithm. This is the same data frame as presented in Fig. 1. b) The smoothness constraint enforced by the binary cost function allows for tracking even when discontinuities are present in the ice-bottom data, as can be seen around the center of the echogram above. c) and d) Data detrending and adaptive pixel intensity weighting allow for weak bed echoes to be accurately tracked.

published in [5] and [8]. Three examples of tracked radar imagery are shown in Fig 9. Example 9a is the original frame from Fig. 1 which contained an ice-free section on the right side where the two layers merge because of the ice mask information. Example 9b shows the smoothness constraint helping bridge a gap through a section of weak signal and 9c and 9d show a section that benefited from the detrending where the whole ice-bottom was generally weak, but still detectable.

It is crucial to note that previous solutions discarded appreciable amounts of data considered of poor quality or in which the bottom was not clearly visible. We have utilized all segments in which ground-truth data were available; specific sections of the tracking results that present large deviations to the reference have significantly shifted the mean error measurement. The results obtained by our solution are significantly improved when the algorithm is applied to more recent radar datasets due to the improvements in the radar systems.

TABLE II
2D IMAGE TRACKING ERROR RESULTS (IN PIXELS)

Error	Viterbi [6]	MCMC [8]	Level-sets [5]	Viterbi (Ours)
Mean	43.1	37.4	6.6	6.0
Median	14.4	9.1	2.1	1.0

Table II shows the results for our modified implementation of the Viterbi algorithm, as well as results for previously proposed solutions, in terms of absolute column-wise difference compared to manually detected ground truth, measured in pixels and averaged between all frames analyzed.

B. 3D imagery

We executed both of our modified Viterbi and TRW-S algorithm implementations on 3D imagery resulting from the 2014 NASA Operation IceBridge Canadian Arctic Archipelago campaign. Previously published results included only 7 frames, whereas these results include all 102 frames from the dataset. The tracked result for Fig. 3 is shown in Fig. 10. A more difficult and interesting result in

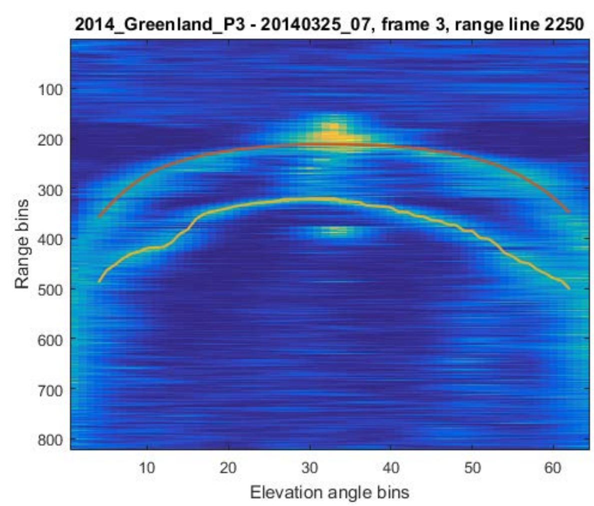

Fig. 10. Example of labeled 3D slice displaying the known ice-surface layer and the ice-bottom layer tracked by our implementation of the TRW-S algorithm. This is the same data frame as presented in Fig. 3.

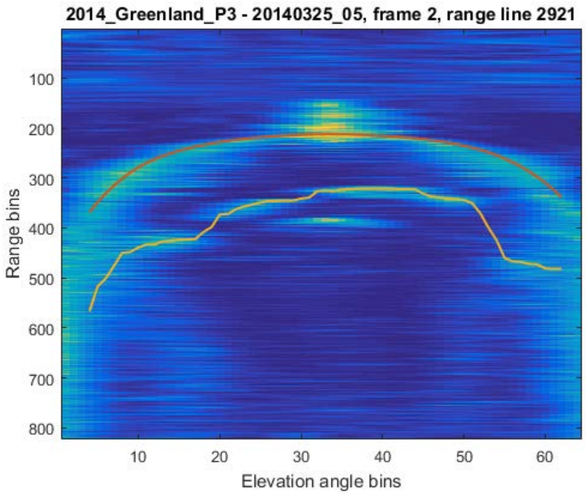

Fig. 11. Example of tracking through ice-bottom data discontinuities on a 3D slice.

shown in Fig. 11 where several discontinuities in the ice-bottom are handled smoothly by the tracker.

The ground-truth against which these results are compared was obtained by manual correction of the results primarily using the TRW-S algorithm. Because the algorithms allow additional ground truth to be passed in, manual ground truth points were added until the bottom layer was tracked in a satisfactory way. If the image quality was too poor to be tracked, then a quality mask was set so that the results for that section of the imagery would not be included in the comparison. The use of the automated trackers to create the manually tracked result is necessary for the 3D dataset due to its large size. For this reason, it is likely that the results presented here are biased towards the results output by the completely automated TRW-S algorithm. This effect does not happen in the case of 2D echograms, as manual tracking of these images is a far more tractable problem.

TABLE III
3D IMAGE TRACKING ERROR RESULTS (IN PIXELS)

Error	Viterbi	TRW-S	Viterbi (Ours)	TRW-S (Ours)
Mean	12.1	9.7	9.8	5.1
Median	2.0	2.0	1.0	0.0

Table III shows the results for our modified implementations of the algorithms, as well as results for the originally proposed implementations, in terms of absolute column-wise difference compared to manually detected ground truth, measured in pixels and averaged between all frames analyzed.

C. *Crossover errors*

TABLE IV
CROSSOVER ERROR RESULTS (IN METERS)

Error	Manually Corrected	TRW-S
Mean	23	26
Median	11	13

Table IV presents the crossover errors where flightlines crossed and two independent measurements were acquired over the same location. We obtain results for both the automated TRW-S algorithm and for the manually-corrected ground-truth data. The dataset used for this calculation is the same as used for the 3D imagery tracking presented above.

Crossover errors can be visualized by overlaying the two (crossing) flightlines of interest in a digital elevation model. Fig. 12 displays the flight paths (green and blue lines) of two data frames from the 2014

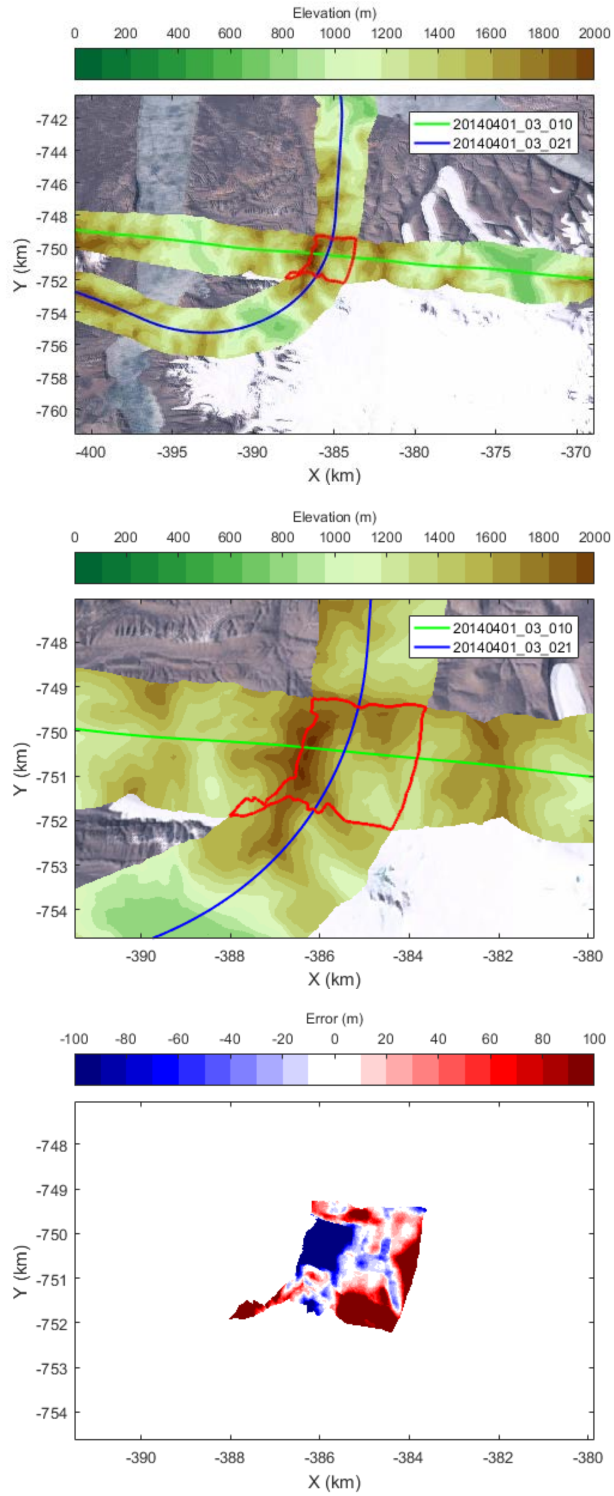

Fig. 12. Crossover visualization and error map.

NASA Operation IceBridge deployment, as well as the swath imaged by each. The region surrounded by a red line is the intersection of the two swaths and represents the data points that were imaged both times. The third section of the figure displays the vertical error between the results obtained by tracking (TRW-S) the ice-bottom layer at the intersection of the two data frames shown.

D. *Geostatistical Analysis*

We perform an examination of the statistical properties of the ice layers after tracking and validation of the results. This analysis is

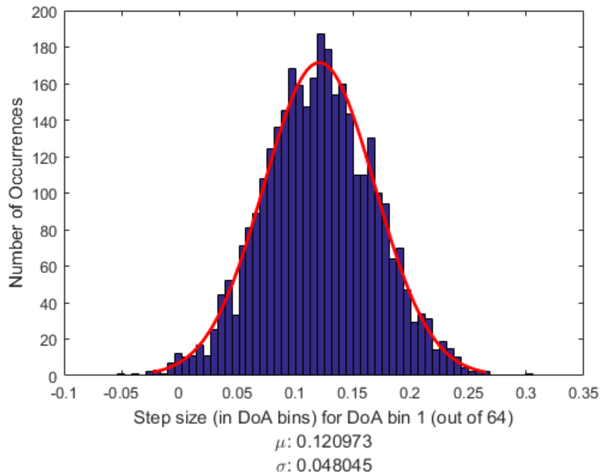

Fig. 13. Histogram of step sizes (in units of direction-of-arrival bins) for bin 1 (out of 64, as shown in Fig. 3). The red line is a fitted Gaussian distribution over the data.

valuable in detecting trends and biases of the detected layers, and has offered clues regarding potential improvements to the cost functions used by the algorithms. We expect it will also be useful in future improvements of the layer tracking technique, in which cost terms may be assigned based on the probability distributions generated by this geostatistical analysis. We compute two distributions, both generated through calculations performed on the 2014 NASA Operation IceBridge dataset.

The first, shown in Fig. 13, is a distribution of direction-of-arrival "step sizes" per unit change in range-bin. "Step size" refers to the horizontal variation, in units of direction-of-arrival bins, between two given layer points. In other words, this is a distribution of direction-of-arrival bin variation of ice-bottom layers when a unit change in range bin index is made.

The second distribution, shown in Fig. 14, contains information regarding average ice thickness per distance to nearest ice-margin. The term "ice-margin" refers to the meeting point between icy and non-icy regions. The nearest ice-margin can be found in either direction-of-arrival or range-line dimensions.

VI. Conclusion

In this work we have demonstrated ice-bottom tracking in 2D SAR images and 3D SAR tomographic images of glacial ice using adaptations of two existing algorithms. The results of these tracking algorithms are compared with previous results and generally perform better due to additional evidence and a better fit to the specific problem of ice imaging. Measurements at crossing points, suggest an accuracy of 23 m in elevation for the 3D SAR images after manual corrections are applied. Geostatistical analysis of the absolute and differenced ice thickness after manual correction suggests relatively smooth probability density functions that may be useful in improving the automated tracker in the future.

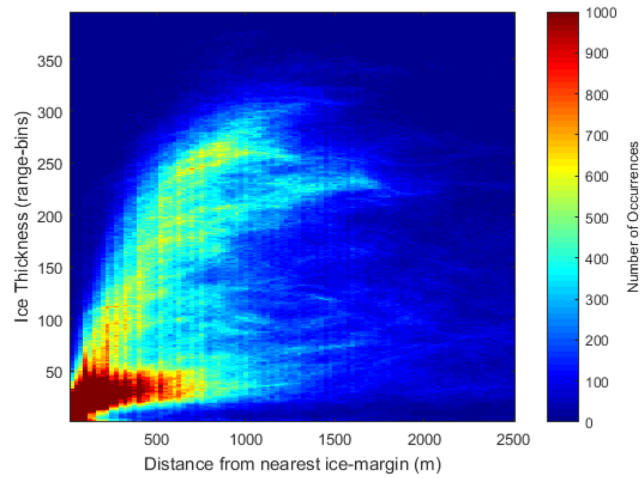

Fig. 14. Two-dimensional histogram of ice thickness (in units of range-bins) as a function of distance from the nearest ice-margin (in units of meters).

VII. References

[1] A. Gilbert, G. Flowers, G. Miller, B. Rabus, W. Van Wychen, A. Gardner, L. Copland, "Sensitivity of Barnes Ice Cap, Baffin Island, Canada, to Climate State and Internal Dynamics," Journal of Geophyiscal Research – Earth Surface, vol. 121, Issue 8, 2016.

[2] F. Rodríguez-Morales, P. Gogineni, C. Leuschen, J. Paden, J. Li, C. Lewis, B. Panzer, D. Gomez-García Alvestegui, A. Patel, K. Byers, R. Crowe, K. Player, R. Hale, E. Arnold, L. Smith, C. Gifford, D. Braaten, C. Panton, "Advanced Multifrequency Radar Instrumentation for Polar Research," IEEE Trans. Geosci. Remote Sens., vol.52, no.5, pp. 2824-2842, 2014.

[3] M. Al-Ibadi, J. Sprick, S. Athinarapu, V. Berger, T. Stumpf, J. Paden, C. Leuschen, F. Rodríguez, M. Xu, D. Crandall, G. Fox, D. Burgess, M. Sharp, L. Copland, W. Van Wychen, "Crossover Analysis and Automated Layer-Tracking Assessment of the Extracted DEM of the Basal Topography of the Canadian Arctic Archipelago Ice-Caps," 2018 IEEE Radar Conference (RadarConf18), Oklahoma City, OK, 2018, pp. 862-867.

[4] C. M. Gifford, G. Finyom, M. Jefferson, Jr., M. Reid, E. L. Akers, and A. Agah, "Automated polar ice thickness estimation from radar imagery", IEEE Trans. Image Process., vol. 19, no. 9, pp. 2456-2469, Sep. 2010.

[5] M. Rahnemoonfar, G. C. Fox, M. Yari and J. Paden, "Automatic Ice Surface and Bottom Boundaries Estimation in Radar Imagery Based on Level-Set Approach," in IEEE Trans. Geosci. Remote Sens., vol. 55, no. 9, pp. 5115-5122, Sept. 2017.

[6] D. J. Crandall, G. C. Fox and J. D. Paden, "Layer-finding in radar echograms using probabilistic graphical models," Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, 2012, pp. 1530-1533.

[7] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260-269, 1967.

[8] S. Lee, J. Mitchell, D. Crandall, G. Fox, "Estimating bedrock and surface layer boundaries and confidence intervals in ice sheet radar imagery using MCMC," International Conference on Image Processing (ICIP), 2014, pp. 111–115.

[9] M. Xu, D. Crandall, G. Fox, and J. Paden, "Automatic Estimation of Ice Bottom Subsurfaces from Radar Imagery," IEEE International Conference on Image Processing (ICIP), 2017.

[10] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," Transactions on Pattern Analysis and Machine Intelligence, vol.28, no. 10, pp. 1568–1583, 2006.

[11] M. Xu, C. Fan, J. Paden, G. C. Fox and D. J. Crandall. "Multi-Task Spatiotemporal Neural Networks for Structured Surface Reconstruction," IEEE Winter Conference on Applications of Computer Vision (WACV), 2018.

[12] J. Paden, T. Akins, D. Dunson, C. Allen, P. Gogineni, "Ice-Sheet Bed 3-D Tomography," Journal of Glaciology, vol.56, no.195, pp. 3-11, 2010.

[13] P. Fretwell, et. al., Bedmap2: improved ice bed, surface and thickness datasets for Antarctica, The Cryo., 7, 375–393, 2013.

[14] M. Morlighem, E. Rignot, H. Seroussi, E. Larour, H. Ben Dhia, D. Aubry, "A mass conservation approach for mapping glacier ice thickness," Geophys. Res. Lett., vol. 38, issue 19, 2011.

[15] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization", The Journal of Machine Learning Research, vol.13, pp. 281-305, 2012.

[16] L. Rabiner, "A Tutorial on HMMs and Selected Applications in Speech Recognition," Proceedings of the IEEE, vol. 77, no. 2, pp. 257-286, Feb. 1989.

[17] Arendt, A., et al., 2012, Randolph Glacier Inventory [v2.0]: A Dataset of Global Glacier Outlines. Global Land Ice Measurements from Space, Boulder Colorado, USA, Digital Media, 2012.