Automated Ice-Bottom 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 , *Senior Member, IEEE*, and Geoffrey Charles Fox

Abstract—Multichannel radar depth sounding systems are able to produce two-dimensional (2D) and three-dimensional (3D) imagery of the internal structure of polar ice sheets. Information such as ice thickness and surface elevation is extracted from these data and applied to research in ice flow modeling and ice mass balance calculations. Due to a large amount of data collected, we seek to automate the ice-bottom layer tracking and allow for efficient manual corrections when errors occur in the automated method. We present improvements made to previous implementations of the Viterbi and sequential tree-reweighted message passing (TRW-S) algorithms for ice-bottom extraction in 2D and 3D radar imagery. These improvements are in the form of novel cost functions that allow for the integration of further domain-specific knowledge into the cost calculations and provide additional evidence of the characteristics of the ice sheets surveyed. Along with an explanation of our modifications, we demonstrate the results obtained by our modified implementations of the two algorithms and by previously proposed solutions to this problem, when compared to 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. Using our modified cost functions and preprocessing routines, we obtain significantly decreased mean error measurements from both algorithms, such as a 47% reduction in average tracking error in the case of 3D imagery between the original and our proposed implementation of TRW-S.

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 these polar topography data helps determine the contribution of these icy bodies 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 sea level [1].

We use Multichannel Coherent Radar Depth Sounder [2] data to form images of the subterranean ice topography. The radar employs a cross-track antenna array where each antenna element is individually sampled. The data from each element is independently pulse compressed to resolve targets in range and synthetic aperture radar (SAR) processed to resolve targets in along-track. Finally, the individual SAR images from each element are combined using array processing to resolve targets in the elevation angle dimension [3]. Data products include two-dimensional (2D) SAR images where only the nadir elevation angle is resolved and three-dimensional (3D) SAR tomographic images where targets from all angles are resolved. In both of these data product formats, 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. Accurate manual tracking of 2D echograms at this scale 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).

Automated ice-layer tracking on 2D radar imagery has received attention from researchers including Gifford *et al.* [4], who proposed edge-based and active-contour-based iterative methods for tracking the interfaces. A distance-regularized level-set technique was proposed by Rahnemoonfar *et al.* [5], while Crandall *et al.* [6] posed this tracking as an inference problem on a probabilistic graphical model, a hidden Markov model (HMM), to incorporate several known constraints of the ice layers, and used the Viterbi algorithm [7] to solve it. A similar, more general model was suggested by Lee *et al.* [8], but they used Markov Chain Monte Carlo (MCMC) to solve the inference problem more accurately. All of the aforementioned

techniques allowed for the ice-surface and ice-bottom layers to be simultaneously detected.

Carrer and Bruzzone [9] propose a technique for extracting internal ice layers in radar imagery using a local scale HMM and performing inference with the Viterbi algorithm. However, this method is not suitable for explicit extraction of the ice-bottom layer because it identifies several similar boundaries within the ice, whereas the solutions proposed in this paper aim to track only the ice-bottom layer while purposely ignoring other boundaries. Smock and Wilson [10] also present a method for identifying the ground surface and other layer boundaries in ground-penetrating radar data, using a reciprocal pointer chain technique designed as an improvement over the standard Viterbi algorithm. However, their task is not specialized to ice sheet layer tracking, and, thus, the models they use are very different.

Specifically for ice-bottom tracking in 3D imagery, Xu *et al.* [11] define an inference problem using a Markov random field (MRF) and apply sequential tree-reweighted message passing (TRW-S) [12] to solve it. The performance is compared to a Viterbi algorithm with modifications specific for the 3D imagery. They later proposed a deep convolutional and recurrent neural network that achieved improved results [13].

In this paper, we present adaptations to the aforementioned HMM-based and MRF-based solutions described in [11], adjusting the cost functions to include additional domain-specific knowledge and evidence regarding polar ice sheets. We apply the adapted algorithms to CReSIS data, and perform an assessment in terms of tracking accuracy, calculated as the average absolute difference between our automated results and manually tracked ground truth data, and compare against previously proposed algorithms. We also check the self-consistency of the 3D algorithms by comparing results where flightlines cross, such that two independent measurements have been made at the same location.

For the solutions proposed in this paper, we make the assumption that the ice-surface layer is known *a priori* since there generally exist accurate ice-surface estimates (such as ArcticDEM [14] and Bedmap2 [15]) based on satellite imagery for surveyed locations. The location of the ice-surface is used as an input to the ice-bottom trackers and defines portions of the cost function as explained in Section III.

The remainder of this paper is organized 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. In Section III, we define our modifications to the original models, and in Section IV we present our algorithms used to perform inference on these models. In Section V, we present and analyze experimental results. In Section VI, we briefly look at a geostatistical analysis of our tracked results. The conclusions are then summarized in Section VII.

II. BACKGROUND

A. 2D Imagery

In a 2D image or echogram (e.g., Fig. 1), the subterranean ice structures are captured along the flight profile. The horizontal
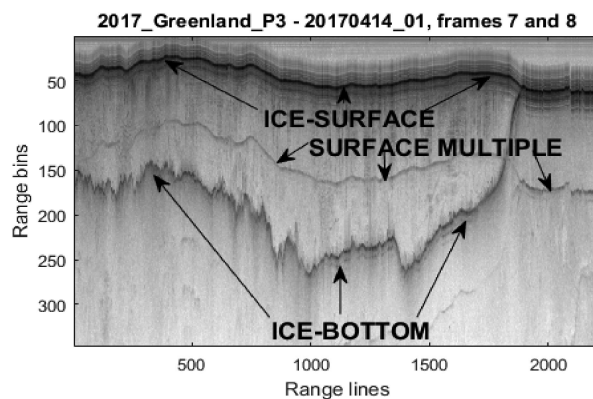

Fig. 1. Example of 2D echogram. Notice that the ice-surface and ice-bottom merge on the right when no ice is present.
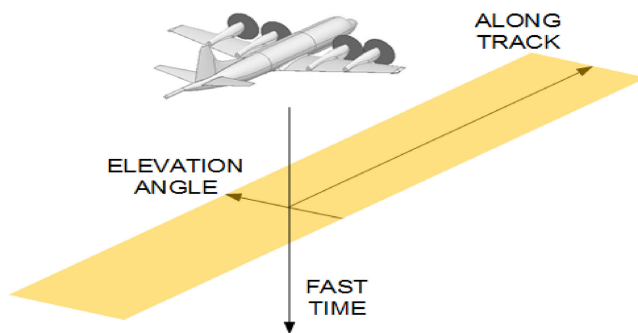

Fig. 2. Illustration depicting the axes of the images relative to the radar platform. The 2D echograms are vertically oriented and lie in the plane formed by the fast time and along-track axes at zero cross track.

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. Fig. 2 illustrates the image axes with respect to the aircraft. The pixel intensity of the radar image is proportional to radar scattering intensity, with darker representing a stronger scattering signal. The array processing algorithm for 2D imagery uses a filter to estimate the intensity of targets in the nadir direction while suppressing targets from off nadir. The minimum variance distortionless response (MVDR) algorithm is used for 2D images in this paper [16].

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 boundary. Notice that these two interfaces merge around range-line 1900, indicating the beginning of a region with no ice. Furthermore, 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 is an artifact that may confuse the tracker and create a false positive, complicating the layer tracking process. A simple solution for mitigating the effects of this undesirable feature is described in Section III.
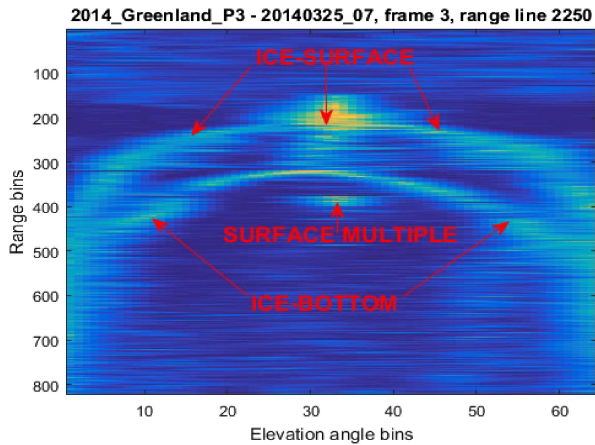
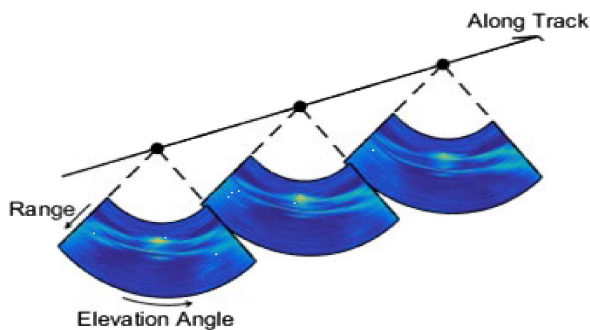Fig. 3.   Example of a 3D image slice. Elevation angle bin 33 is the nadir.

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

B. 3D 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 2D echogram where each row corresponds to a range-bin. Fig. 3 shows a sample image slice, and Fig. 4 shows how the slices fit together relative to the radar platform coordinate system. The multiple signal classification (MUSIC) array processing algorithm is used to generate the images [17]. The color of each pixel represents the MUSIC cepstrum, which is loosely related to how likely a scatterer is present. In Fig. 3, yellow indicates a larger cepstrum value, which is associated with increased likelihood, and blue indicates a lower cepstrum value. The first surface multiple is also present in most slices of the 3D imagery but tends to be less detrimental to automated tracking as compared to 2D imagery because it is seen only in the nadir elevation angle bin.

C. Layer Tracking Framework for 2D and 3D Imagery

While both formats of radar imagery capture the subterranean ice sheet structure 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: It tends to vary
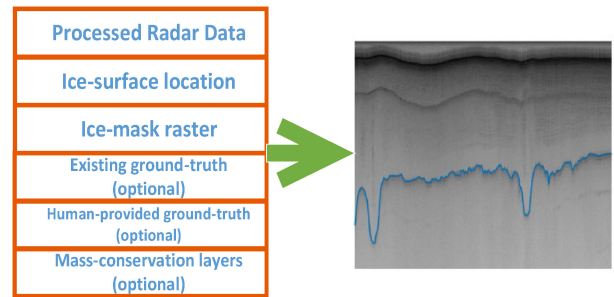

Fig. 5.   Diagram illustrating the main inputs of the automated layer-tracking systems. The location of the ice-bottom layer, shown here as a continuous blue line overlaid on a 2D echogram, is the output for both 2D and 3D trackers.

smoothly in both elevation angle and along-track dimensions. On the other hand, no "third dimension" is available for the 2D 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 2D case, the reduced dimensionality of the data allows for an exact solution to the cost minimization to be found using the Viterbi algorithm (although less information is supplied to the minimization, and so the result is generally worse than the 3D case [11]). In the 3D case, the algorithm with the best results in this study exploits the layer correlation in both dimensions.

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 be the correct boundary position in each column of the image matrix. In a physical sense, this is the same as assuming that the subterranean ice structure contains no overhangs or cave-like features from the perspective of the radar. Fig. 5 provides a summary of the other inputs and constraints to the layer tracking software.

III. Graphical Model for Layer Tracking

First consider the 2D labeling problem, where the goal is to trace the location of the ice bed. More concretely, given a N_r-by-N_x radar echogram intensity image matrix I, our goal is to identify, for each column (or range line) c of I, the corresponding row coordinate s_c through which the ice bed passes. The tracking output $S = (s_1, s_2, \ldots, s_{N_x})$ is, thus, in the form of an N_x-dimensional vector, where N_x is the number of range-lines in the input image and $s_i \in \{1, \ldots, N_r\}$ where N_r is the number of rows or range bins. This ensures that exactly one row will be selected as the ice-bottom label for every column of the image. A simple approach is to define a model that encourages each column to simply choose the row with the strongest intensity within that column, in other words,

$$s_c = \arg \min_r \{-I(r, c)\}. \tag{1}$$

The overall cost function for N columns is

$$E(S) = \sum_{c=1}^{N} -I(s_c, c). \tag{2}$$

However, radar echograms are noisy observations of the true ice structure, and this simple technique will often lead to layer boundaries that are physically implausible (e.g., that are discontinuous). Successful layer tracking, thus, requires balancing the noisy evidence from the echogram with known prior information about the typical properties of polar ice. In previously published work, Crandall *et al.* [6] poses the echogram layer-tracking problem as an inference problem on a statistical graphical model, which is designed for combining evidence from multiple sources. For both 2D and 3D imagery, the authors assign a *unary cost ψ_U* to every valid pixel, which represents the cost for the ice-bottom layer to pass through that pixel, just as in the simple model above. This unary cost is augmented to encode additional constraints, such as marking a labeling as invalid (infinite cost) if the boundary passes above the known ice-surface layer, or if it is at the same position as the ice-surface layer at a point where a nonzero ice thickness is known to exist. In addition, a *binary cost* function assigns a cost ψ_B to every pair of adjacent columns (range-lines). This binary cost enforces "smoothness" of the layer boundary by penalizing sharp discontinuities (very different rows assigned to adjacent columns).

This formulation suggests an alternative optimization problem that should find an S to minimize the sum of all of these unary and pairwise costs:

$$E(S) = \sum_{c=1}^{N} \psi_{\mathrm{U}}(s_c, c) + \sum_{p=(c_i, c_j)}^{p \in P} \psi_{\mathrm{B}}(s_{c_i}, s_{c_j}) \qquad (3)$$

where P is the set of all 2-tuples that represent neighboring columns.

This combination of unary and binary cost functions defines a HMM, which can be solved efficiently and exactly with the Viterbi algorithm, as we discuss below. Note that the form of (3) follows the minimization of the negative logarithm of the probabilities which is equivalent to the maximization of the posterior probability. This form of the inference can also be viewed as a cost minimization problem.

In the case of a N_r-by-N_d-by-N_x 3D image, the tracking output is a N_d-by-N_x matrix where each element represents the output for a given column and N_d represents the number of elevation angle bins in the 3D image matrix. The value of N_r still represents the number of range-bins, and the value of N_x still represents the number of range-lines which is equal to the number of N_r-by-N_d slices that compose the 3D space. In this case, the pairwise term enforces smoothness between adjacent columns in the slice (elevation-angle bins) and between adjacent columns between slices (range-lines). One can visualize these constraints as a grid graph since the label of every column now depends on its neighbors both within and across slices. This is a MRF, which, unlike an HMM, cannot be solved efficiently using Viterbi. In fact, MRF inference is NP-hard in general, which means that no efficient *exact* algorithm is known to exist. Fortunately, very good *approximate* algorithms are known, including the TRW-S algorithm used by [11].

Next, we present an expanded explanation of this cost assignment process, as well as our proposed modifications to the Viterbi and TRW-S algorithms used in [11].

A. Unary Cost Function

The unary cost function ψ_U assigns a cost for labeling an individual column with a given row label. Intuitively, this cost function encourages the estimated layer boundary to pass through high-response areas of the echogram, while also enforcing physical constraints about ice layers that can be expressed within single columns of the echogram. In particular, our unary cost function consists of five types of terms.

1) A radar return strength term that encourages layers to lie along response peaks of the echogram.
2) A hard constraint on the ice bed that forces it to be below the ice surface unless the ice thickness is known to be zero.
3) Soft constraints on ice thickness that encourage the bed to be not too close to the surface.
4) A term that strongly encourages the ice bed to be consistent with known ground truth points, if available.
5) A term that weakly encourages agreement with weaker evidence about ice thickness (e.g., based on mass conservation models [18]). The unary cost for any column c is just a sum of these five terms:

$$\psi_U(s, c) = \psi_{\mathrm{SURF}}(s, c) + \psi_{\mathrm{SINC}}(s, c) + w_{\mathrm{GT}}\psi_{\mathrm{GT}}(s, c)$$
$$+ \psi_{\mathrm{EXTRA}}(s, c) + w_{\mathrm{REP}}\psi_{\mathrm{REP}}(s, c) \qquad (4)$$

where the w variables are weights that are tuned via hyperparameter optimization.

We now describe each of these terms in detail.

1) Radar Return Strength, $\psi_{\boldsymbol{SINC}}$: Previous work [11] measures the sum 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 toward the edges of the template. Because the term measured the squared distance to the template, a peak response in the imagery with the *exactly same* intensity values produces the lowest cost.

We modified the previously proposed template term in order to better use the dynamic range of the imagery. A problem with using the squared distance to the template is that the peak intensity from the ice-bottom layer varies and larger intensities generally indicate a better measurement (since these correspond to greater signal-to-noise ratios). This meant that values with a larger peak intensity than the template would actually be penalized. To better handle peak intensity variability, we now use a correlation operation that multiplies the template with the image:

$$\psi_{\mathrm{SINC}}(s, c) = -\sum_{p \in T} I(s+p, c)\mu(p) \qquad (5)$$

where I is the input image and $p \in T = \{-5, -4, -3, \ldots, 5\}$ refers to the pixel index of the correlation function, and $\mu(p)$ is the correlation function which is now fixed to $\mathrm{sinc}(p/3.33)$, 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 truncated sinc waveform was chosen because it is symmetric and has a single peak in the
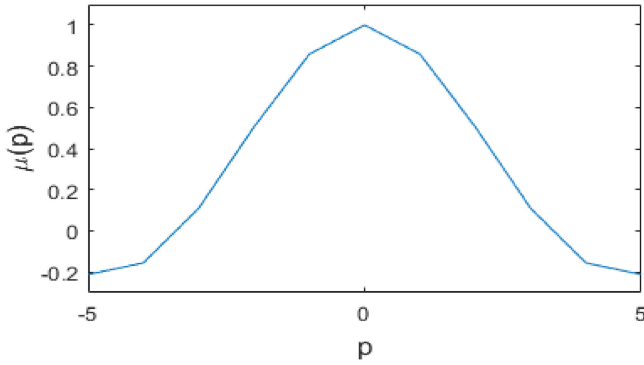
Fig. 6. Values of the correlation function $\mu(p)$ used in the unary cost calculation ψ_{SINC}.

center. Improved performance is likely possible by training on the waveform shape.

2) Surface and Thickness Constraint, ψ_{SURF}: An obvious physical constraint is that all points of the ice-bottom layer must be at or below the ice-surface and that the two layers may only be at the same range if no ice is present at that position. Since ice-mask datasets, which show where ice is present, are available for most regions surveyed by CReSIS (e.g., Randolph Glacier Inventory [19]), we define ψ_{SURF} to:

1) merge the ice-bottom to the ice-surface where there is no ice (i.e., $M_{\text{ice}}(c) = 0$);
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);
3) have no effect when more than a certain distance away from the ice margin.

In particular,

$$\psi_{\text{SURF}}(s,c) = \begin{cases} \infty, & s > r_{\text{surf}}(c) \text{ or } (s \neq s_{\text{surf}}(c) \\ & \text{and } M_{\text{filt}}(c) = 0) \\ 0, & s_{\text{surf}}(c) > s > s_{\text{surf}}(c) + M_{\text{filt}}(c) \end{cases}$$
(6)

where s represents the row index of the pixel of interest, $s_{\text{surf}}(c)$ is the row index of the ice-surface layer at column c, and $M_{\text{filt}}(c)$ is a filtered version of the binary ice-mask $M_{\text{ice}}(c)$. This filtered version is obtained by the following:

1) eroding the mask by 2 pixels;
2) filtering with a weighted boxcar window;
3) setting values above 90 to infinity.

The filter for 2D imagery is scaled by $(90/3.7)$ and is applied only in the along-track dimension. The filter for 3D imagery is scaled by $(90^{0.5}/3.7)$ and is applied in the elevation angle dimension and then again in the along-track dimension effectively creating a 2D filtering of the mask. The following shows the effect of these steps on an ice edge boundary for 2D imagery. In this case, the ice mask is a vector corresponding to each column of the 2D image.

1) Erosion of the mask by 2 pixels: THE binary mask $M_{\text{ice}} = [0\ 0\ 1\ 1\ 1\ 1\ 1]$ becomes $[0\ 0\ 0\ 0\ 1\ 1\ 1]$.
2) Filtering with a boxcar window: $[0\ 0\ 0\ 0\ 1\ 1\ 1]$ becomes $[0\ 0\ 24\ 49\ 73\ 97\ 122]$.

3) Infinity threshold at 90: $[0\ 0\ 24\ 49\ 73\ 97\ 122]$ becomes $M_{\text{filt}} = [0\ 0\ 24\ 49\ 73\ \infty\ \infty]$.

This process takes into account the fact that a relatively smooth transition is expected between icy and nonicy regions of the terrain, and restricts the ice-bottom to a range of values near the ice margin. The choice of the particular filter coefficients and threshold were manually chosen and have not been tuned. A geostatistical analysis of actual ice thickness distributions near the ice margin would likely produce improved results and is discussed in Section VI.

3) Ice Surface Repulsion, ψ_{REP}: 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. 1, 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 ice-bottom.

A shifted exponential decay function was chosen to ensure a smooth cost increase as a function of proximity to the ice-surface, as can be seen in Fig. 8. This term is calculated as

$$\psi_{\text{REP}}(s,c) = \begin{cases} 0, & \Delta_Y > \alpha_{\text{MSD}} \\ \alpha_{\text{MC}} * e^{-\lambda * \Delta_Y} - \alpha_{\text{MC}} * e^{-\lambda * \alpha_{\text{MSD}}}, & \text{otherwise} \end{cases}$$
(7)

where $\Delta_Y = s_{\text{surf}}(c) - s$ is the vertical pixel distance between $s_{\text{surf}}(c)$, the row index of the ice-surface layer at column c, and the pixel of interest s, and λ is a manually chosen exponential decay constant. ψ_{REP} is different from ψ_{SURF} as it will gradually increase the unary cost of pixels as they vertically approach the given ice-surface layer regardless of the ice-mask value of the column of interest, whereas ψ_{SURF} enforces a hard constraint that is either positive infinity or zero and effectively restricts the range of allowed values for s.

4) Ground Truth Constraints, ψ_{GT}: Known layer positions may be available for some columns of an echogram. To account for potential small inaccuracies in the ground truth, we do not use a hard constraint, but instead use a squared distance term that strongly encourages the estimated row for each column to be close to the ground truth:

$$\psi_{GT}(s,c) = (s - s_{\text{GT}}(c))^2$$
(8)

where s represents the row index of the pixel of interest and s_{GT} is the row index of the ground truth point. When ground truth is not available, this term of the unary cost function is simply set to 0. The values of the ψ_{GT} term are demonstrated in Fig. 7.

These ground truth layer positions can come from a variety of different sources. 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 found by finding all intersections between the flight path of interest and flight paths of previous surveys. Frequently, a given location will have been imaged and labeled before, and the point in which the new flightline crosses the old
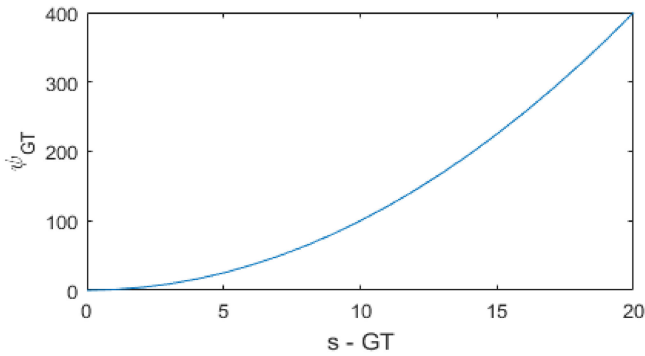
Fig. 7. Unary cost of every pixel in the input image is increased according to the squared vertical distance to ground truth points in the same column of the image matrix, if they exist.
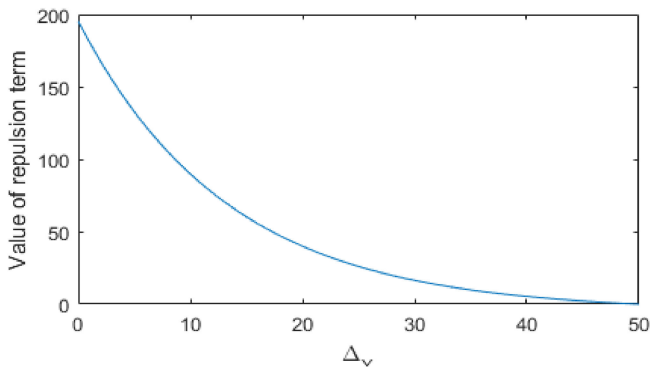

Fig. 8. Shifted exponential decay of the ψ_{REP} term as a function of the vertical distance Δ_Y between the pixel of interest and the ice-surface layer. The selected parameter values are $\alpha_{\mathrm{MSD}} = 50$, $\alpha_{\mathrm{MC}} = 200$, and $\lambda = 0.075$.

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 paper, 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.

5) Ground Truth Estimates, ψ_{EXTRA}: Another potential source of evidence is *a priori* estimates of the ice-bottom that can be used as weak evidence by the tracking algorithms. These estimates can be handled similarly to the ground truth constraints above, except that they are added to the cost function with a lower weighting to reflect their lower reliability. These estimates can be obtained, for example, from existing ice thickness models based on ice flow dynamics and mass conservation [18].

B. Binary Cost Function

The binary cost is a function of the row labels assigned to adjacent columns and is designed to encourage smoothness of estimated layer boundaries, increasing the likelihood that transitions, which generate smoother layers, will be selected by assigning 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 is in a cylindrical coordinate system native to the radar sounder processing, this "flat" surface did not represent a flat surface in Cartesian space. A flat surface in Cartesian coordinates curves downward toward 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, we modified the cost function to assign 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 exact shape of a flat ice-bottom in Cartesian space, which would need to account for ice refraction from a nonflat ice-surface layer. The new binary cost between two adjacent columns c_i and c_j is given by

$$\psi_B\left(c_i, c_j, t\right) = w_B * \left(s_{c_i} - s_{c_j} - \left(S_{\mathrm{surf}}\left(c_i\right) - S_{\mathrm{surf}}\left(c_j\right)\right)\right)^2 \tag{9}$$

where s_{c_i} and s_{c_j} are row indices assigned to adjacent columns, and $S_{\mathrm{surf}}(c)$ is the row assigned for the ice surface for a column c. A scaling factor w_B is used to define the weight of this smoothing term. In the 2D imagery case, this modification is not as helpful since a flat (i.e., perfectly horizontal) ice boundary in a 2D echogram approximately depicts flat topography of the surveyed terrain, because it does not account for the change in radar wave propagation speed in ice versus air. However, although it is a small correction, there is often a correlation between the range-slope of the ice-surface and the shape of the ice-bottom layer [20], [21]; thus, this cost term is still used. A promising approach for removing this assumption from both tracking frameworks involves using empirically discovered column-to-column transition probability distributions based on a geostatistical analysis of the manually corrected data (see Section VI).

C. Data Preprocessing

The 2D image intensity exhibits a strong dependence on depth in ice due to ice loss and spherical spreading loss which often leads to a very large dynamic range between the surface and bottom return intensities. In our modified version of the Viterbi, we apply a simple detrending routine that normalizes the mean intensity of each row or range bin. This helps the tracker in areas where the bed echo is weak.
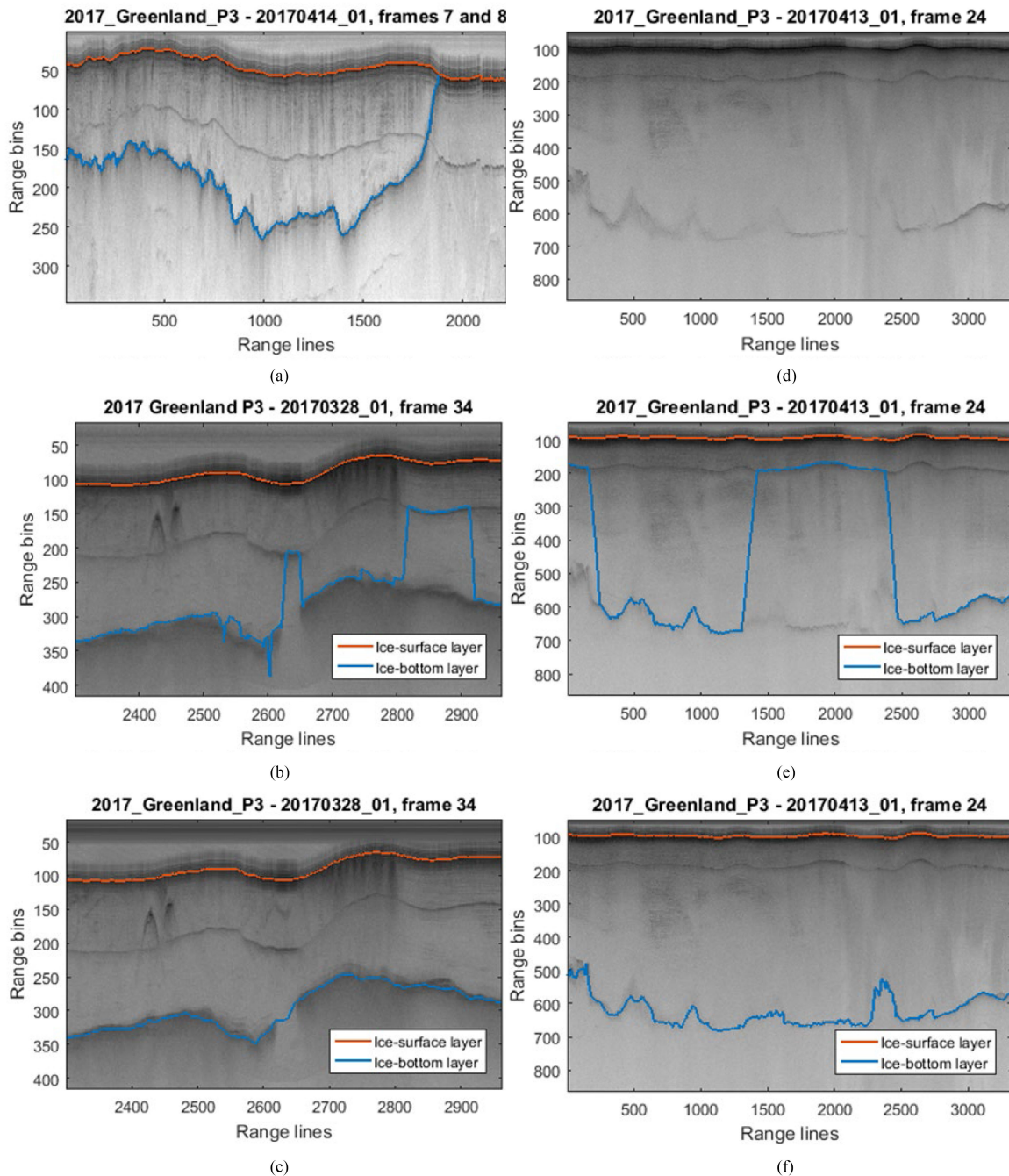
Fig. 9. (a) Example of labeled 2D echogram displaying the known ice-surface layer and the ice-bottom layer tracked by our implementation of the Viterbi algorithm. This is the same data frame as presented in Fig. 1. Comparison between (b) old Viterbi and (c) new Viterbi implementations. The surface multiple suppression and 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. (d) Echogram with weak bottom echoes and a comparison of (e) old Viterbi and (f) new Viterbi tracking. Data detrending and surface multiple suppression allow for weak bed echoes to be accurately tracked.

Without detrending, clutter near the ice-surface is often so strong that the ice-bottom layer tracker may jump to this signal despite the surface-repulsion and layer smoothness constraints enforced by the unary and binary cost functions of the tracking algorithms. Fig. 9 be example of this. The first proposed Viterbi solution [6], which worked with 2D images only, used image gradients and cumulative max gradients to handle the dynamic range.

While the 2D images are estimates of scatterer intensity from MVDR, the 3D images output the MUSIC cepstrum and a similar detrending procedure is not necessary because MUSIC produces a muted dynamic range. In [11], which dealt with the 3D MUSIC images, a simple thresholding technique was used to reduce the dynamic range. With this approach, every pixel of the input image with a value greater than a certain manually tuned threshold was made equal to the threshold value to reduce

the difference between the pixel values of the ice-surface and ice-bottom layers to prevent the automated tracker from selecting the (often) stronger ice-surface return as the ice-bottom. The problem with thresholding is that the shape and strength of the return are distorted by the thresholding and occasionally the ice bottom would exceed this threshold and be clipped as well. The thresholding step was removed for the 3D images in our modified approach.

As previously mentioned, an undesirable feature present in the 2D imagery is the surface multiple, which is caused by a ringing of the radar signal between the ice-surface and the aircraft. This is also seen in the 3D imagery, but the tracking is not affected. To mitigate the effect of the surface multiple in 2D imagery as a false positive to the algorithms, we employed a simple method of smoothing the input image around the areas in which the surface multiple is located. This was achieved by replacing from 20 rows above to 20 rows below the surface multiple with a blurred version of the image. The image is blurred with a 2D Gaussian filter with standard deviation equal to 50 pixels and kernel size equal to 201. It is possible to estimate the location of the first surface multiple by doubling the two-way travel time of the ice-surface. We have experimentally found that this approach does not significantly slow down the tracking process or decrease tracking accuracy even if the ice-bottom layer is located within the rows to which the filtering was applied, due to the smoothness and continuity constraints enforced by the unary cost function. We did, however, find the rate of mislabeling the ice-bottom due to the surface multiple to have been largely decreased.

Additionally, previous 2D tracking efforts performed layer tracking on the 2D echograms of individual data frames. This sometimes resulted in lower quality results near the edges of the data frames because evidence from contiguous frames was not being considered. For this reason, we horizontally concatenated 2D data frames before passing them to the Viterbi algorithm so that all adjacent data frames (usually entire flights) are tracked at once. This also increases the probability that the frames being processed will include automated ground truth from crossovers, although this tends to have a relatively local effect on improving performance. The 2D echogram presented in Fig. 1 is composed of two horizontally concatenated data frames (see title of figure).

D. Parameter Optimization

The weights and parameters in the cost functions (such as the maximum sensory distance α_{MSD} and maximum cost α_{MC} of the ψ_{REP} term described above) affect the accuracy of the automated trackers, so their values were tuned for the best possible performance of the algorithms. We used multistage (multiresolution) grid-search and random-search [22] parameter optimization techniques, 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 for large parameter spaces where not all variables

TABLE I
PARAMETERS USED IN COST FUNCTION CALCULATIONS

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

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 different combination of optimal parameters was found for each. 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, as well as the scaling factor w_B for the binary cost function. The optimal values found from this process are shown in Table I.

IV. ALGORITHMS APPLIED TO LAYER TRACKING

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

Regarding time complexity, in the general case, the computation time of both TRW-S and Viterbi is quadratic in the number of possible values of the hidden states (i.e., number of rows of the echogram in our case) because the binary cost term must consider all possible pairs of transitions. However, the fact that our binary cost function is a Gaussian allows for the messages to be computed much more efficiently, in time linear to the number of hidden states, using generalized distance transforms (also called the minimum convolution operation) [6], [23]. On a typical desktop machine, the Viterbi algorithm takes about 2 s to process a 50-km 2D image. For 3D imagery, the same frame takes 20 s for our Viterbi implementation and around 1800 s for our TRW-S implementation. Since the processing of radar data is trivially parallelized by assigning an independent job to each data frame, the additional computation time of TRW-S is not an issue for our application.

A. 2D Imagery and the Viterbi Algorithm

For the 2D imagery, we follow the solution proposed by Crandall *et al.* [6] of solving the HMM inference problem using the Viterbi algorithm [7] to perform exact inference. Viterbi is an efficient dynamic programming algorithm for 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. Our modified implementation is most like the HMM in [11] and the description of the modifications is relative to this implementation rather than the implementation in [6].

B. 3D Imagery and the TRW-S Algorithm

As described above, in order to take advantage of the strong correlation between consecutive slices of 3D imagery, Xu *et al.* [11] proposes an MRF model. An MRF is similar to an HMM, in that it consists of unary and pairwise terms but an important difference is that the graphical structure of pairwise terms in an HMM forms a chain, whereas an MRF may have an arbitrary graph structure. In the case of 3D ice imagery, the graphical structure is a grid graph, where every column has four neighbors (two within the slice and two across slices). Unfortunately, while exact inference on HMMs is efficient using the Viterbi algorithm, inference on MRFs is NP-hard in the general case. Fortunately, a number of *approximate* inference algorithms have been developed. Xu *et al.* [12] use sequential TRW-S, which is the de facto state-of-the-art in approximate MRF inference. TRW-S is an iterative message-passing algorithm similar to the belief propagation algorithm. In belief propagation, messages are exchanged among the nodes of the MRF. The messages contain likelihood distributions that reflect each variable's beliefs about the labels of its neighbors, based on its own evidence and the messages it has received in past iterations. Local evidence, thus, propagates from neighbor to neighbor, spreading across the entire graph given enough iterations. After iterating is complete, each node chooses its label based on its local evidence and the last set of messages it received from its neighbors. TRW-S is also based on message passing, but the messages are exchanged along carefully constructed subsets of the graph, and final belief distributions are obtained by combining estimates across these subsets together. Although TRW-S is still an approximation algorithm, these enhancements give TRW-S better convergence and accuracy bounds than the classic belief propagation algorithm.

Because of the interslice message passing capability, this method is capable of preventing discontinuities in both along-track and elevation angle dimensions during the layer reconstruction. While the intraslice message passing procedure performs in a similar fashion to the Viterbi algorithm by propagating evidence to its neighboring pixels to the left and right (elevation-angle dimension), the interslice 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 [11] so that the message passing of the current iteration's messages along the elevation-angle dimension is now always performed outward from nadir, rather than switching from left-to-right and right-to-left 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 elevation-angle 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 direction of the current iteration's message was changed to be always outward from nadir, in such way that the nadir column asserts the greatest influence on the final result.

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 [11], we found that 50 iterations usually produce 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 3D matrices can be passed in.

As expected, when applied to 3D 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. 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 paper. Note that such an approach would still be weaker than TRW-S because it would make decisions in a sequential manner and would consider slices sequentially instead of simultaneously. For example, a bad decision in the first slice would negatively impact inference on subsequent slices because that bad estimate would be fed in as ground truth. In contrast, TRW-S considers all evidence simultaneously, meaning that ambiguity in the first slice may be resolved by evidence in later slices.

V. RESULTS AND DISCUSSION

A. 2D Imagery

We tested our modified Viterbi routine on 2D data from several NASA Operation IceBridge campaigns. The algorithm received no manual aid of any kind, and the only ground truth points provided were the aforementioned automatically acquired crossovers. Three examples of tracked radar imagery are shown in Fig 9. Fig. 9(a) is the original frame from Fig. 1 that contained an ice-free section on the right side where the two layers merge because of the ice mask information. Fig. 9 compares the (b) old [6] and (c) new Viterbi tracking. This example demonstrates the surface multiple suppression and smoothness constraints helping to bridge a gap through a section of no signal. Fig. 9(d)–(f) shows another comparison between the (e) old and (f) new algorithms where detrending and surface multiple suppression helped the new algorithm track the whole ice-bottom. Finally, Fig. 10(a) shows a comparison between the ground truth and our automated result.

To compare the results to the other published algorithms, we tested our modified Viterbi routine on the 2009 NASA Operation IceBridge Antarctica campaign, the same dataset used by the authors of Rahnemoonfar *et al.* [5], Crandall *et al.* [6], Lee *et al.* [8]. We did not rerun the other algorithms for the 2D imagery; rather we compare to the results published in [5], [6], and [8]. It is crucial to note that the reported results in [5] and [8] discarded appreciable amounts of echograms considered to be too poor quality for tracking. In our results, we have utilized all range lines
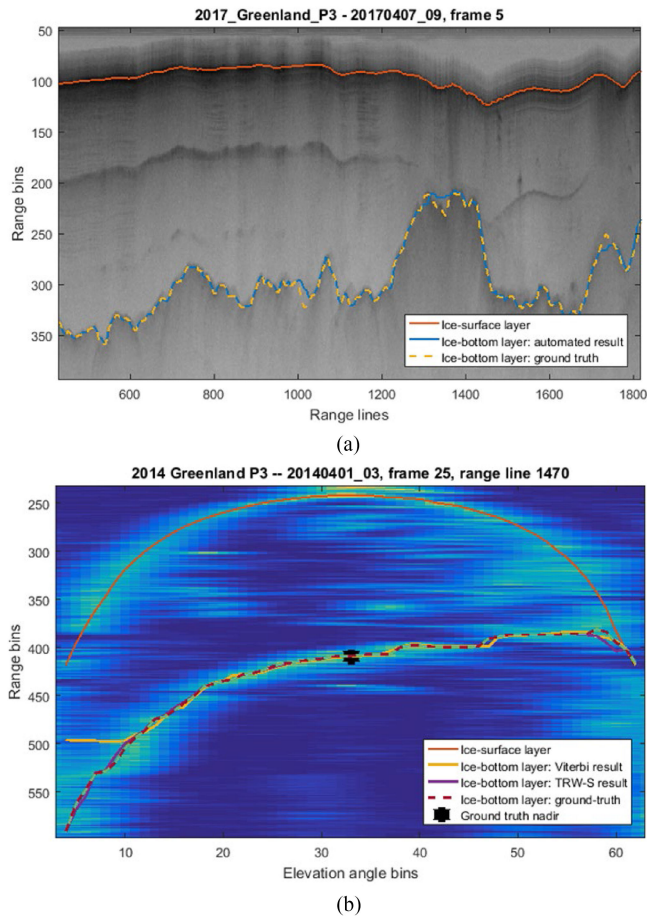
(a)

(b)

Fig. 10. Comparison of (a) 2D Viterbi tracking and (b) 3D TRW-S tracking and manually selected ground truth.

Fig. 11. Example of tracking through discontinuities on a 3D slice.

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

Error	Viterbi [6]	MCMC [8]	Level-sets [5]	Viterbi (Ours)
Mean	32.0	37.4	6.6	6.2
Median	-	9.1	2.1	1.0

No median error was calculated in [6].

where ground truth labels were available. It is likely that these poor quality sections increased the mean error measurement of the new Viterbi algorithm relative to [5] and [8].

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. For accuracy and precision, we define a true positive for a particular image column to be a tracked layer within three range bins of the ground truth. A true negative and a false positive column are then any column for which this is not the case. Therefore, the accuracy and precision are identical and equivalent to the percentage of correctly labeled columns which we found to be 85%.
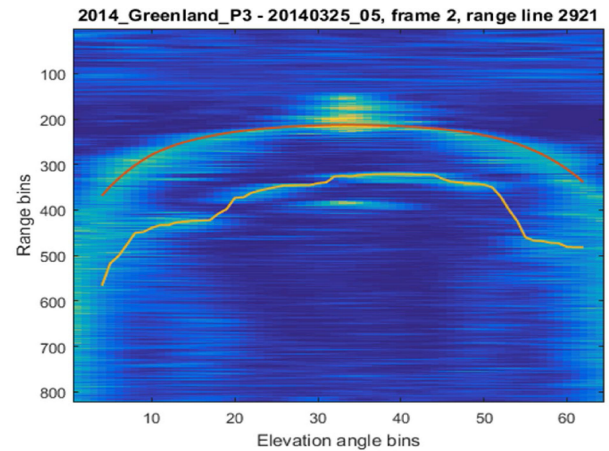
It should be noted that the three previous solutions [5], [6], [8] in Table II are able to automatically extract *both* ice-surface and ice-bottom layers from a 2D echogram. Providing a layer tracking algorithm with the location of the ice-surface boundary, such as we propose, simplifies the tracking framework and strengthens the location constraints for the ice-bottom. For the ice surface used by our Viterbi algorithm on the 2D images tracked in this paper, the tracking is done by a fully automated threshold tracker aided by satellite-based ice-surface location estimates. In the 2D imagery, the ice surface is usually the dominant scatterer and is the first reflection, so the simple automated threshold tracker tends to do very well.

Furthermore, all of the three previous solutions included here [5], [6], [8] make use of less information than our proposed solution. In all three, the *a priori* location of the ice-surface and the ice-mask raster are not used by the algorithms; the only input requirements of these techniques consist of the radar echogram itself and a set of tunable hyper-parameters such as weights and scaling factors. The original Viterbi implementation [6] allows for ground truth points to be optionally added, either automatically or by a human annotator, for both ice-surface and ice-bottom layers. This is useful in cases where the automated tracker makes a mistake: a human reviewer may then be able to quickly add a small number of ground truth points and rerun the algorithm. The published MCMC [8] and level-sets [5] methods do not allow for corrections or improvements to be made by a human annotator via the addition of ground truth points.

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 seven frames, whereas these results include all 102 frames from the dataset. A comparison of the ground truth and the modified TRW-S algorithm is shown in Fig. 10(b). A more difficult and interesting result is shown in Fig. 11 where several discontinuities in the ice-bottom are handled smoothly by the tracker.

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

Error	Viterbi [11]	TRW-S [11]	Viterbi (Ours)	TRW-S (Ours)
Mean	12.1	9.7	9.8	5.1
Median	2.0	2.0	1.0	0.0

TABLE IV
CROSSOVER ABSOLUTE ERROR RESULTS (IN METERS)

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

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 manually 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 toward the results output by the completely automated TRW-S algorithm. This is not the case for 2D echogram results, as manual tracking of these images is a far more tractable problem.

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. The accuracy and precision of our TRW-S result is 87% using the same definition as for 2D imagery.

C. Crossover Errors

Table IV presents the crossover errors where flightlines crossed and two independent measurements were acquired over the same location. The two independent measurements allow the consistency of the approach to be analyzed. We present 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 image tracking presented above. Crossover errors can be visualized by overlaying the two (crossing) flightlines of interest in a digital elevation model. Fig. 12(a) and (b) displays the flight paths (green and blue lines) of two data frames from the 2014 NASA Operation IceBridge deployment, as well as the swath imaged by each. The region surrounded by the red line is the intersection of the two swaths and represents the data points that were imaged both times. Fig. 12(c) displays the vertical error between the results obtained by tracking the ice-bottom layer at the intersection of the two data frames shown. Table IV represents the average
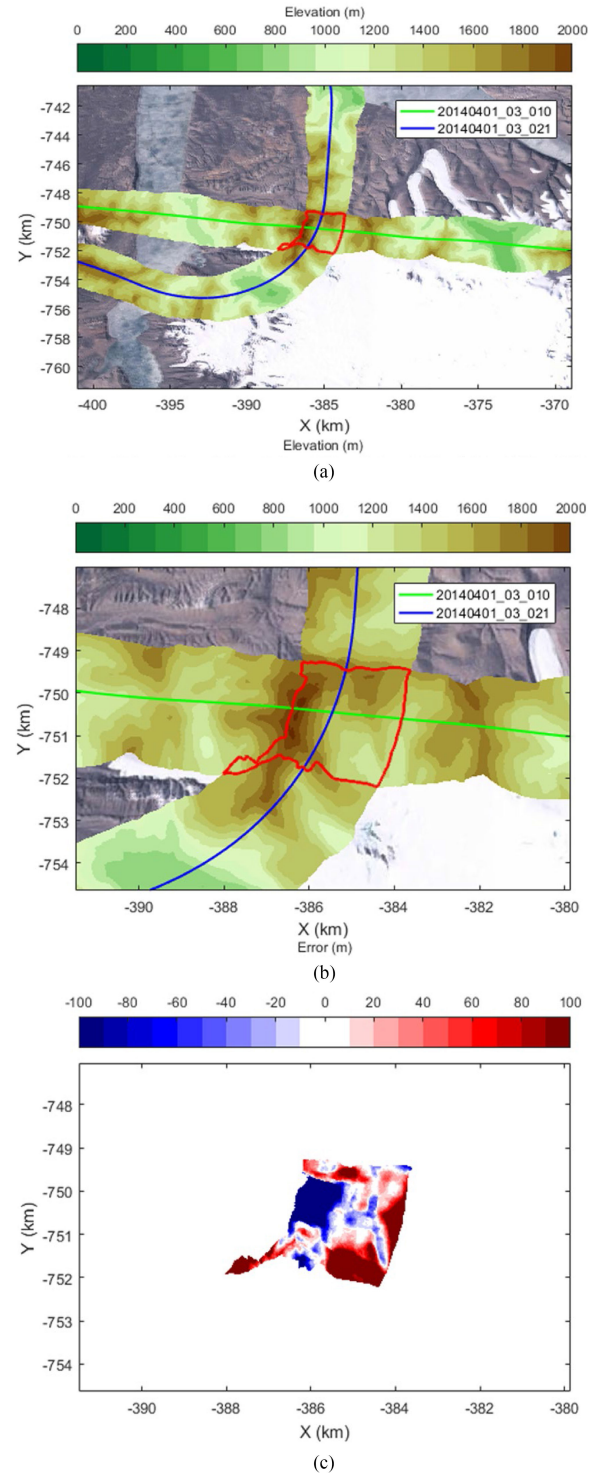

Fig. 12. Crossover visualization and error map.

of all 20 crossovers in the dataset; the mean absolute error for Fig. 12(c) is the worst among all the crossovers and is 71.6 m.

VI. GEOSTATISTICAL ANALYSIS

We perform an examination of the statistical properties of the ice layers after tracking and validation of the results. This
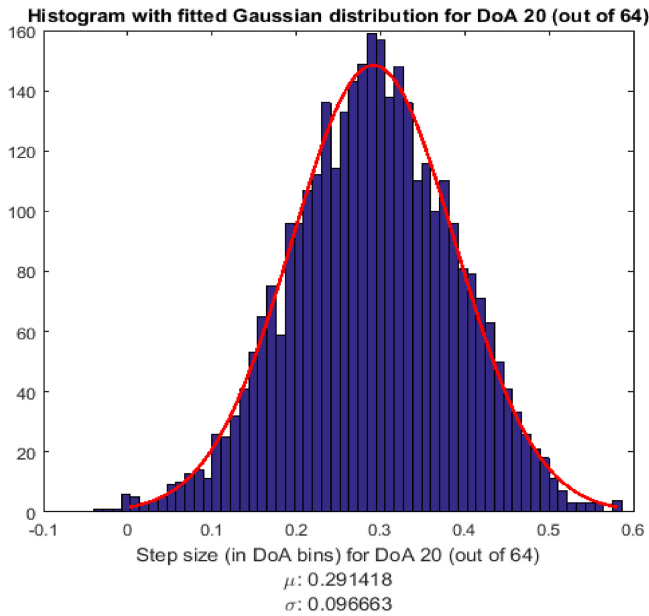
Fig. 13. Histogram of step sizes (in units of direction-of-arrival or DoA bins) at elevation bin 20 (out of 64, as shown in Fig. 3) when the range bin is increased by one. Vertical axis indicates number of occurrences. The red line is a fitted Gaussian distribution over the data.
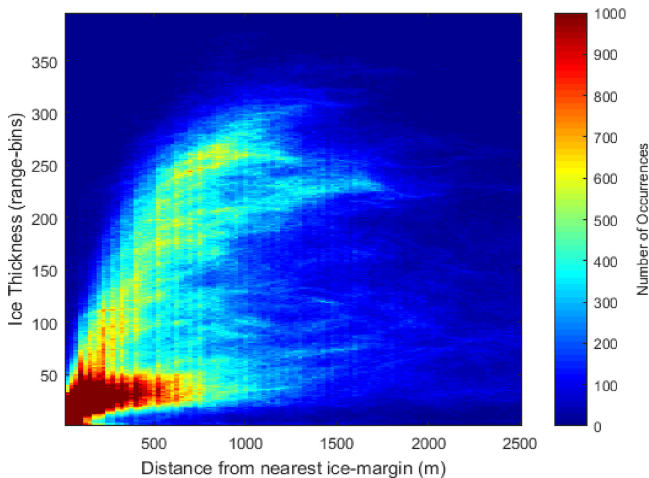

Fig. 14. 2D histogram of ice thickness (in units of range-bins) as a function of distance from the nearest ice-margin (in units of meters).

analysis is 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 that 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 versus distance to nearest ice-margin.

The term "ice-margin" refers to the meeting point between icy and nonicy regions. The distance to the nearest ice-margin is calculated using the Euclidean distance from each point on the surface to the nearest nonicy region.

VII. CONCLUSION

We have demonstrated ice-bottom layer tracking in 2D SAR images and 3D SAR tomographic images of polar ice sheets using modified versions of two existing layer tracking solutions based on the Viterbi and TRW-S algorithms from [11] and compared our results to existing ice bottom tracking algorithms. Based on the results obtained, the modified Viterbi algorithm is optimal for tracking the ice bottom in 2D imagery and the modified TRW-S is optimal for 3D surface reconstruction. The Viterbi algorithm efficiently finds the exact global minimum for the formulated HMM framework, and the TRW-S algorithm iteratively performs statistical inference on the MRF formulated from unary and binary cost functions while considering constraints between adjacent slices due to its 2D message passing properties.

The improvements proposed here arise from refinements made to the unary and binary cost functions used in the original publications, which allow for the integration of further domain-specific knowledge and sources of evidence. Additional automated preprocessing steps are also applied on the 2D radar echograms, such as data detrending, concatenation of adjacent data frames, and blurring of the first surface multiple from the image, which increase the accuracy of the tracker in adverse scenarios such as noisy data and very weak ice bed returns.

The results obtained after our proposed modifications are compared with the results from the original solutions using Viterbi and TRW-S, as well as two more existing solutions using MCMC and level-set techniques. Regarding the results presented for the 2D imagery, it is noted that the previous 2D solutions track both the ice-surface and ice-bottom simultaneously while using fewer sources of evidence. Also, the published results for [5] and [8] excluded some imagery that was hard to track, whereas we have included all imagery in our Viterbi results. Even with these harder to track images included, the modified Viterbi algorithm outperforms these other algorithms. Nonetheless, a direct comparison of all algorithms on the same image set is needed to provide a more precise comparison of the algorithms.

We qualify the results from our modified implementation of the TRW-S algorithm for 3D data when compared against manually corrected data because the manually corrected data is obtained from manually aiding the same TRW-S algorithm. This was done due to the infeasibility of manually labeling several tens of thousands of individual slices with no automated

assistance, and for this reason, the results are likely biased towards a lower mean error for TRW-S.

A self-assessment of the results output by the modified TRW-S algorithm was performed at crossing points and suggests a mean difference of 26 meters in elevation for the 3D SAR images. For context, the manually corrected data exhibits a mean difference of 23 m in the same crossover points.

Finally, a geostatistical analysis of the absolute and differenced ice thickness after manual correction is also proposed. The initial results obtained from it suggest relatively smooth probability density functions that may be useful in improving automated trackers in the future.
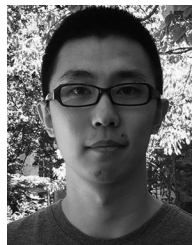
REFERENCES

[1] A. Gilbert *et al.*, "Sensitivity of Barnes ice cap, Baffin Island, Canada, to climate state and internal dynamics," *J. Geophys. Res., Earth Surface*, vol. 121, no. 8, 2016.

[2] F. Rodríguez-Morales *et al.*, "Advanced multifrequency radar instrumentation for polar research," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2824–2842, May 2014.

[3] M. Al-Ibadi *et al.*, "Crossover analysis and automated layer-tracking assessment of the extracted DEM of the basal topography of the Canadian Arctic Archipelago ice-caps," in *Proc. IEEE Radar Conf.*, 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," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5115–5122, Sep. 2017.

[6] D. J. Crandall, G. C. Fox, and J. D. Paden, "Layer-finding in radar echograms using probabilistic graphical models," in *Proc. 21st Int. Conf. Pattern Recognit.*, 2012, pp. 1530–1533.

[7] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[8] S. Lee, J. Mitchell, D. Crandall, and G. Fox, "Estimating bedrock and surface layer boundaries and confidence intervals in ice sheet radar imagery using MCMC," in *Proc. Int. Conf. Image Process.*, 2014, pp. 111–115.

[9] L. Carrer and L. Bruzzone, "Automatic enhancement and detection of layering in radar sounder data based on a local scale hidden Markov model and the Viterbi algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 962–977, Feb. 2017.

[10] B. Smock and J. Wilson, "Reciprocal pointer chains for identifying layer boundaries in ground-penetrating radar data," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2012, pp. 602–605.

[11] M. Xu, D. Crandall, G. Fox, and J. Paden, "Automatic estimation of ice bottom subsurfaces from radar imagery," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 340–344.

[12] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1568–1583, Oct. 2006.

[13] M. Xu, C. Fan, J. Paden, G. C. Fox, and D. J. Crandall, "Multi-task spatiotemporal neural networks for structured surface reconstruction," in *Proc. IEEE Winter Conf. Appl. Comput. Vision*, 2018, pp. 1273–1282.

[14] ArcticDEM. [Online]. Available: https://www.pgc.umn.edu/data/arcticdem/. Accessed: Sep. 2, 2018.

[15] P. Fretwell *et al.*, "Bedmap2: Improved ice bed, surface and thickness datasets for Antarctica," *Cryosphere*, vol. 7, pp. 375–393, 2013.

[16] J. Li *et al.*, "High-altitude radar measurements of ice thickness over the Antarctic and Greenland ice sheets as a part of operation ice bridge," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 742–754, Feb. 2012. doi: 10.1109/TGRS.2012.2203822.

[17] J. Paden, T. Akins, D. Dunson, C. Allen, and P. Gogineni, "Ice-sheet bed 3-D tomography," *J. Glaciol.*, vol. 56, no. 195, pp. 3–11, 2010.

[18] M. Morlighem, E. Rignot, H. Seroussi, E. Larour, H. Ben Dhia, and D. Aubry, "A mass conservation approach for mapping glacier ice thickness," *Geophys. Res. Lett.*, vol. 38, no. 19, 2011. [Online]. Available: https://doi.org/10.1029/2011GL048659

[19] A. Arendt *et al.*, "Randolph glacier inventory [v2.0]: A dataset of global glacier outlines," in *Global Land Ice Measurements from Space*. Boulder, CO, USA: Digital Media, 2012.

[20] N. F. McIntyre, "Antarctic ice-sheet topography and surface–bedrock relationships," *Ann. Glaciol.*, vol. 8, pp. 124–128, 1986.

[21] J. De Rydt, G. H. Gudmundsson, H. F. J. Corr, and P. Christoffersen, "Surface undulations of Antarctic ice streams tightly controlled by bedrock topography," *Cryosphere Discuss.*, vol. 6, no. 6, pp. 4485–4516, Oct. 2012.

[22] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.

[23] P. F. Felzenszwalb and D. P. Huttenlocher, "Distance transforms of sampled functions," *Theory Comput.*, vol. 8, no. 1, pp. 415–428, 2012.

Victor Berger received the B.Sc. degree in computer engineering and the M.S. degree in computer science from the University of Kansas, Lawrence, KS, USA, in 2017 and 2019, respectively.

He then joined the Center for Remote Sensing of Ice Sheets as a Graduate Research Assistant. In 2019, he successfully defended his thesis, entitled "Probabilistic graphical techniques for automated ice-bottom tracking and comparison between state-of-the-art solutions." His research interests include a field deployment in 2018 to assist with the processing efforts of CReSIS polar radar data collected over Antarctica as part of NASA's Operation IceBridge.

Mingze Xu received the M.S. degree in computer science from Indiana University, Bloomington, IN, USA, in 2014, and the B.E. degree in software engineering from Jilin University, Changchun, China, in 2012. He is currently working toward the Ph.D. degree in computer science with Indiana University.

His research interests include the area of computer vision and deep learning, especially on action and activity recognition, first-person (egocentric) vision, image and video segmentation, and embodied visual recognition.

Mohanad Al-Ibadi received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Technology in Baghdad, Baghdad, Iraq, in 2006 and 2009, respectively, and the Ph.D. degree in electrical engineering from the University of Kansas, Lawrence, KS, USA, in 2019.

He is currently a Faculty Member with the Engineering Technical College of Najaf at Al-Furat Al-Awsat Technical University, Iraq. He was a Graduate Research Assistant with the Center for Remote Sensing of Ice-Sheets. His current research interests include the area of radar array signal processing, mainly developing algorithms for estimating and tracking the bottom of an ice-sheet using nonlinear filtering techniques, wideband direction of arrival estimation, and wideband model order estimation.

Shane Chu received the B.A. degree in mathematics from the University of Kansas, Lawrence, KS, USA, in 2018. He is currently working toward the Ph.D. degree in computer science with Washington University in St. Louis, St. Louis, MO, USA, where he is researching the design of motif discovery algorithms for protein-DNA interactions.

He was previously a Research Assistant with the Signal and Data Processing Group, Center for Remote Sensing of Ice Sheets.

John Paden (S'95–M'06–SM'19) received the M.Sc. and Ph.D. degrees in electrical engineering from the University of Kansas (KU), Lawrence, KS, USA, in 2003 and 2006, respectively, studying radar signal and data processing for remote sensing of the cryosphere.

After graduating, he joined Vexcel Corporation, Boulder, CO, USA, a remote sensing company, where he worked as a Systems Engineer and a SAR Engineer for three and a half years before rejoining the Center for Remote Sensing of Ice Sheets (CReSIS), KU, in early 2010 to lead the signal and data processing efforts.

Dr. Paden was the recipient of the American Geophysical Union Cryosphere Early Career Award in 2016 and three NASA group achievement awards for radioglaciology work. He is currently a Research Faculty Member at CReSIS.

David Crandall received the Ph.D. degree in computer science from Cornell University, Ithaca, NY, USA, in 2008, and the B.S. and M.S. degrees in computer science and engineering from Pennsylvania State University, State College, PA, USA, in 2001.

He is an Associate Professor with the School of Informatics and Computing, Indiana University. His research interests include computer vision, machine learning, and data mining.

Dr. Crandall is the recipient of a National Science Foundation CAREER Award and a Google Faculty Research Award.

Geoffrey Charles Fox received the Ph.D. degree in theoretical physics from Cambridge University, Cambridge, U.K., where he was Senior Wrangler.

He is currently a Distinguished Professor of Engineering, Computing, and Physics with Indiana University, Bloomington, IN, USA, where he is the Director of the Digital Science Center. He previously held positions at Caltech, Syracuse University, and Florida State University after being a Postdoc with the Institute for Advanced Study at Princeton, Lawrence Berkeley Laboratory, and Peterhouse College Cambridge. He has supervised the Ph.D. of 73 students and published around 1300 papers (more than 500 with at least ten citations) in physics and computing with an h-index of 78 and more than 35 000 citations. He is involved in several projects to enhance the capabilities of Minority Serving Institutions. He has experience in online education and its use in MOOCs for areas such as data and computational science.

Dr. Fox is a Fellow of APS (Physics) and ACM (Computing) and works on the interdisciplinary interface between computing and applications.