A Deep Active Contour Model for Delineating Glacier Calving Fronts

Konrad Heidler [ID], *Student Member, IEEE*, Lichao Mou [ID], Erik Loebel [ID], Mirko Scheinert [ID], Sébastien Lefèvre [ID], *Senior Member, IEEE*, and Xiao Xiang Zhu [ID], *Fellow, IEEE*

Abstract— **Choosing how to encode a real-world problem as a machine learning task is an important design decision in machine learning. The task of the glacier calving front modeling has often been approached as a semantic segmentation task. Recent studies have shown that combining segmentation with edge detection can improve the accuracy of calving front detectors. Building on this observation, we completely rephrase the task as a contour tracing problem and propose a model for explicit contour detection that does not incorporate any dense predictions as intermediate steps. The proposed approach, called "Charting Outlines by Recurrent Adaptation" (COBRA), combines convolutional neural networks (CNNs) for feature extraction and active contour (AC) models for delineation. By training and evaluating several large-scale datasets of Greenland's outlet glaciers, we show that this approach indeed outperforms the aforementioned methods based on segmentation and edge-detection. Finally, we demonstrate that explicit contour detection has benefits over pixel-wise methods when quantifying the models' prediction uncertainties. The project page containing the code and animated model predictions can be found at https://khdlr.github.io/COBRA/.**

Index Terms— **Active contours (ACs), edge detection, glacier front, Greenland, uncertainty.**

I. INTRODUCTION

RECENT years have seen rapid warming in the polar regions, which has led to an exceptionally large mass loss of the Greenland ice sheet [1]. This loss of ice mass

translates into global sea level rise and can cause feedback effects that further increase the warming of the Arctic [2]. Closely monitoring the Greenland ice sheet is therefore of paramount importance. About half of this ice mass loss is generally attributed to glacier dynamics, like dynamic imbalance and increased discharge. The remaining half is attributed to negative surface mass balance, which mostly stems from an increase in surface melt [3].

Following the rapid changes in air and sea temperature, glacier dynamics in these regions are changing quickly. One essential indicator for dynamic changes of marine-terminating glaciers is the calving front, which is the boundary line of the glacier from which ice bergs calve off. In order to better understand the glaciological processes and provide more accurate constraints for glacier modeling, detailed monitoring of the glaciers' calving fronts is necessary. With the ever-growing availability of satellite remote sensing data, monitoring glaciers at a large scale with high temporal frequency has become possible, but requires automated methods. Therefore, recent years have seen rapid advances in applying machine learning for glacier monitoring, which will be explored in more detail in Section II-A.

With the rise of deep learning methods in remote sensing, the predominant method of approaching this task has been via *semantic segmentation*. In this formulation, each pixel is assigned a label that corresponds to either the *glacier* class or the *sea* class. Given the large number of studies on semantic segmentation in computer vision, the methods and models for this task are well understood and provide decent results when applied to calving front detection. However, these methods require postprocessing steps to extract the actual calving front from the segmentation masks.

Noting that segmentation is only a proxy for the actual task of calving front detection, and neither the sea nor the inward glacier area is of actual interest for calving front detection, the field has seen a recent trend toward edge detection methods. By combining computer vision methods for pixel-wise edge detection with the aforementioned segmentation task, predictions are thus greatly improved [4], [5].

The goal of this study is to provide a new angle on this task. Picking up the trend toward edge detection, we propose to completely move away from pixel-wise prediction architectures and rethink the task from the ground up. The desired final prediction format for calving front detection is a vectorized polyline, which is a data format that is well-suited for downstream analysis and modeling applications. Therefore,
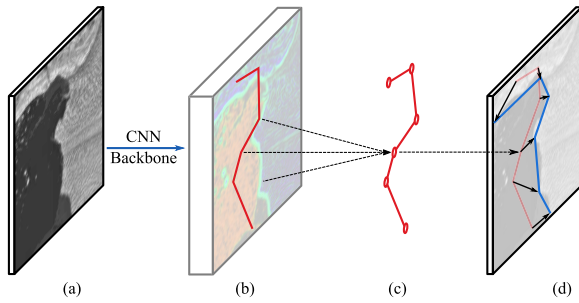
Fig. 1. High-level overview of our deep AC model for delineating calving fronts. (a) First, the backbone network takes the input image and derives feature maps. (b) Then, a sample is taken from these feature maps at the position of each vertex. (c) These features are evaluated by the Snake Head which predicts offsets for each vertex. (d) Finally, the offsets are applied to update the contour. This process is repeated multiple times.

we are looking to build a model that directly outputs the calving fronts in this desired format instead of recovering the vectorized contour from intermediate predictions. By radically redesigning the neural network architecture, we are able to move away from pixel-wise classifiers and instead arrive at a model that directly predicts the calving front as a polyline.

This approach has several theoretical benefits over representing the desired output by a dense, pixel-wise mask.

1) As the predictions are already in a vector format, there is no need for complicated postprocessing pipelines like with pixel-wise approaches.
2) By its very design, the model will learn to focus on the actual object of interest, the calving front.
3) Looking closer into the application, explicit contour prediction provides a natural way of encoding prior knowledge into the network. In pixel-wise detection frameworks, the network may predict undesired outputs like disconnected line segments. By directly predicting an explicit contour, such issues are eliminated.
4) Explicit contours make more efficient use of computational resources. A sequence of vertices takes fewer parameters than a dense mask.
5) Finally, the vectorized representation allows for better quantification of model uncertainty as the joint probability distribution of a sequence of vertices is easier to model than that of pixel-wise masks.

Convinced by these theoretical considerations, we set out to develop a calving front detection model that directly predicts the desired contours, as shown in Fig. 1. Contour-based approaches for the segmentation of regions in natural images have been extensively studied in the form of *active contours*, which are also called *Snakes* [6].

In order to provide robust and stable calving front predictions for downstream applications, such as glaciological studies and models, it is important to quantify the reliability of the model's predictions. Therefore, we also explore the question of uncertainty quantification in calving front detection. Experimental results using the Monte Carlo (MC) dropout method [7] across different models suggest that uncertainty quantification with contour-based models can indeed bring benefits over pixel-wise models.

Overall, we summarize the goals and contributions of this study by the following points.

1) We rephrase the task of the automated glacier calving front detection from a segmentation task to a contour detection task and show that deep active contour (AC) models are a feasible approach to solving this task.
2) We develop a specialized deep AC model for the delineation of glacier calving fronts which outperforms both pixel-based approaches as well as existing deep AC models. The effect of the design decisions is validated through extensive ablation studies.
3) We explore the benefits of contour-based methods for uncertainty quantification compared to pixel-wise methods.

II. RELATED WORK

In order to place our work into the context of existing research, we provide a brief overview of existing calving front detectors, as well as methods for explicit edge predictions.

A. Detecting Calving Fronts in the Deep Learning Era

Given the strong performance of deep learning-based methods for calving front detection, traditional vision methods have largely become insignificant for this task [5], [8]. Therefore, this section focuses on deep learning-based methods. Here, most approaches formulate the task as a variant of sea–land segmentation. In this formulation, a semantic segmentation network is used to separate the image into land and ocean classes [8]. There is a considerable number of well-tested segmentation architectures, like UNet [9], which is a strong baseline for most segmentation tasks. Even without any changes to the network itself, this approach can yield satisfactory results for calving front detection, which has been shown in previous studies for both the Greenland ice sheet [10] and the Antarctic ice sheet [11]. Seeing this strong baseline performance of the UNet, Periyasamy et al. [12] show that the performance of such a model can be greatly improved by tweaking network components, like normalization layers, the loss function, or dropout rate.

Further progress in this field was made by extending the UNet model or exploiting the advances of more recent segmentation model architectures. For example, Loebel et al. [13] add more layers and thereby increase the number of down- and upsampling steps. This enlarges the spatial context that the network can consider for its decisions and therefore leads to better predictions. Following recent advances in transformer-based model architectures, Holzmann et al. [14] enhance the UNet model with attention gates to improve the interpretability of the model and better understand the learning process. Another newer neural network architecture that has successfully been adopted for calving front detection is DeepLabv3+ [15]. Zhang et al. [16], Cheng et al. [4], and Gourmelon et al. [17] bring in ideas from this architecture to obtain more accurate delineations of glacier calving fronts. Finally, it is also possible to combine image classification and segmentation [18], which can lead to more robust results.

Recently, there appears to be a trend toward models that approach calving front detection by extending or even replacing semantic segmentation with edge detection methods.

By focusing on the boundary between the two classes rather than the areas of sea and glacier, these models are encouraged to learn features that are informative of the calving front rather than features of the sea and glacier areas.

Wavelet transforms are one possible approach that makes use of the abrupt changes in texture between glacier and sea to determine the location of the calving front [19]. Davari et al. [20] use a different transformation, namely the Euclidean distance transformation, and train a network that predicts each pixel's distance to the calving front instead of a binary class. Great potential lies especially in the combination of segmentation and edge detection. Both HED-UNet [5] and the calving front machine (CALFIN) [4] choose this approach to outperform models that focus on only one of these two aspects.

Contrary to these models, our approach is not to predict pixel-wise segmentation or edge masks, but instead to explicitly predict a contour parameterized by a fixed number of vertices.

B. Explicit Edge Prediction

The idea of explicitly parameterizing contours in an image was pioneered quite early in the history of computer vision by Kass et al. [6]. They proposed *active contours* or *Snakes*, which evolve from an initial contour by iteratively minimizing an energy functional. By design, this functional takes its minimum when the contour coincides with the desired boundary in the image. Using Radarsat data, this approach has been shown to work for the delineation of the Antarctic coastline on a coarse scale [21]. The main drawback of conventional AC methods is the fact that they are limited to single-channel imagery without any natural extension to multichannel imagery. Furthermore, they are sensitive to local image contrast and the results depend highly on the initialization of the contour.

As automatic feature extraction is a strong suit of deep learning models, the idea of combining ACs with deep learning is not a new one. Rupprecht et al. [22] introduced a deep AC model that works by first predicting a 2-D offset field that points from each pixel toward the closest boundary point. An initial contour will then evolve along this offset field until it converges. However, this method is not end-to-end trainable as it relies on the intermediate offset field and no gradients flow through the actual curve evolution. While this approach can work on multichannel imagery, it still suffers from a strong dependence on contour initialization.

In an effort to introduce an end-to-end trainable deep AC model, Peng et al. [23] proposed to make not only the feature extraction part learnable, but the contour evolution step as well. Their model, termed *Deep Snake* first derives feature maps using a convolutional backbone network and then samples the features at the position of each vertex. From these sampled features, a 1-D convolutional neural network (CNN) then predicts the offsets for each vertex. Like with conventional AC models, this process is then iterated to refine the predictions.

As one of the most recent models in this line of research, deep attentive contours (*DANCE*) [24] improves on the Deep Snake idea by introducing an "edge attention map," which influences the speed of the snake evolution. While vertices far from the target boundary are evolving quickly, the update speed for points closer to the boundary is slowed down.

Inspired by these advances, our goal is to develop an AC model for the task of calving front detection. The existing models address the computer vision task of instance segmentation, where objects in an image are locally segmented. For calving front delineation, however, one global line between glaciers and the ocean is needed. This disparity and further differences, like the general shapes of the objects of interest, call for a completely different network architecture as well as changes to loss functions and the network training protocol.

III. Deep Active Contour Models for Calving Front Delineation

When approaching the task of calving front detection, we first take a look at how a human would proceed in solving the task. In discussions with experts and when annotating calving fronts ourselves, one central observation is the order in which different areas in the scene are addressed.

To a human annotator, it does not make much difference whether they are told to trace the edge between two objects in an image or fill in the areas that both objects occupy. In both cases, they will usually start out by tracing the boundary area between the two classes with minute attention to detail. When asked to perform segmentation, the remaining areas are then filled in with broad strokes in a second step. So while humans will approach both tasks in a similar fashion, a focus on the boundary appears to be the more natural way to formulate the task of calving front detection.

For a neural network model, the way a task is formulated changes everything. As these models are trained to minimize some loss function, the final performance is determined by how well a low loss value translates to good performance on the actual task. For instance, it has been observed that the cross-entropy loss used for training semantic segmentation models can lead to blurry edges between the classes due to the fact that each pixel contributes equally to the final loss, no matter its position in relation to the objects in the image. This implies that a model can minimize most of its loss by correctly classifying the simpler pixels that lie in the interior of the objects of interest. In turn, the model will spend less attention on the pixels near the boundary, which are much more important for solving the task [5], [25].

Phenomena like these are likely the reason that calving front delineation has recently seen a shift toward improving these segmentation models by including pixel-wise edge detection tasks. By putting more focus on the edges, the model is forced to learn how to better distinguish the classes in these critical areas [4], [5]. Instead of combining segmentation with edge detection in a pixel-wise framework, we take a more radical approach in this study. By completely eliminating the semantic segmentation aspect and focusing only on the edges, it is possible to reformulate the task in such a way that it does not require pixel-wise classifications. Instead, the model will directly output a vectorized contour.
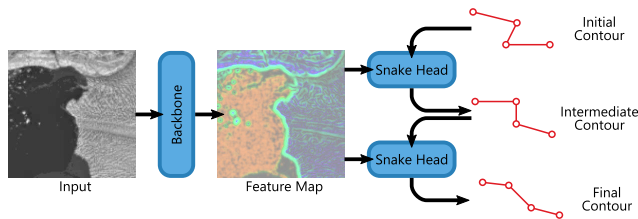
Fig. 2. Architecture overview of our model. Note that while this diagram shows only two iterations of the Snake Head, the number of iterations is actually an arbitrary hyperparameter, which we set to four for our experiments.

A. General Model Architecture

Inspired by the ideas of Peng et al. [23], we develop a deep contour model for the delineation of glacier calving fronts. As is common in many recent computer vision models, our model consists of two main components which perform different subtasks in order to solve the overall task together. The *backbone* is a general-purpose 2-D CNN which is used to extract semantically valuable features from the input imagery. The second component is a *prediction head*, which makes use of the backbone's features to derive the final network predictions. In our model, the prediction head takes the role of the AC iteration. Therefore, it will take a contour and the backbone's feature maps as its inputs, and update the contour to better match the desired boundary. Due to this functional similarity, we call this component the *Snake Head*. The overall architecture of the network is visualized in Fig. 2. Notably, this framework can be trained end-to-end, as all components are fully differentiable.

For the backbone, multiple feature extractors were evaluated. Initial experiments with standard ResNet backbones produced unsatisfactory results. This leads us to believe that while ResNets are a strong backbone for many vision tasks, they are likely not optimal for deep AC models. In search of a better-suited backbone network, the Xception backbone [26] used by Cheng et al. [4] in their study of Greenland's glaciers proved to be a very capable backbone for remote sensing of glaciers that transfers well to deep AC models.

B. Snake Head

The central challenge in predicting contours from an image is the fact that input and output are represented in different dimensionalities. While the input image is represented by a 2-D grid of pixels, the contour that the model should output is given as a sequence of vertices, which is 1-D. The idea of AC models is to start with an initial contour and then iteratively update this contour based on the image values at each vertex. Conceptually, deep AC models do nearly the same thing. However, they do not directly sample the image values but instead, sample the values from the feature map derived by the backbone network.

After sampling the backbone features at the vertex positions, the Snake Head predicts an offset for each vertex, which represents how the vertex needs to be shifted so that the entire sequence of vertices can better represent the true contour. This is achieved by using a 1-D CNN. While conventional, 2-D CNNs pass information between adjacent pixels, the 1-D CNN used in the Snake Head passes information between adjacent
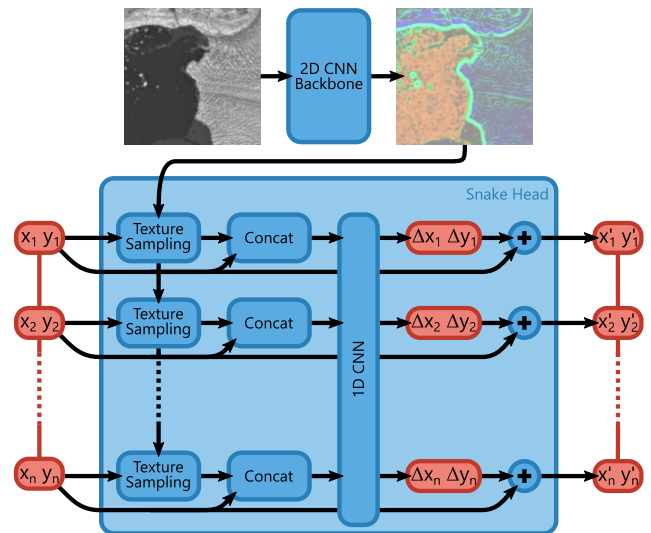

Fig. 3. Detailed view of the Snake Head. After the feature maps are sampled at the vertex positions, the vertex coordinates are concatenated to the vertex features. The 1-D CNN then predicts offsets for each vertex. These offsets are added to the input coordinates to obtain the Snake Head's output.

vertices of the contour. In order to enable the passing of information between vertices that are far away from each other, we stack multiple such convolutional layers. The receptive field of the Snake Head is further increased by using dilated convolutions. In our model, the Snake Head is therefore given as a stack of dilated convolutions. We set the sequence of dilation rates to 1, 3, 9, 9, 3, and 1, as similar setups have proven to be successful at capturing low- and high-frequency features in signal processing tasks [27].

In order for the Snake Head to gain some spatial reasoning capabilities, we also include the vertex coordinates as additional input features for the Snake Head CNN. This allows the model to not only ensure a homogeneous spacing of the output vertices but also to learn some prior assumptions on the shape of the calving fronts. The overall working mechanism of the Snake Head is shown in Fig. 3.

To translate the iterative nature of the AC method, we apply the Snake Head multiple times with the shared weights to obtain more refined predictions. Starting with the initial contour, the Snake Head samples features at the vertex positions, calculates and applies the offsets, and repeats the process. Compared to conventional AC models, which can take dozens [21] or even hundreds [28] of iterations to converge, the deep AC model converges to satisfactory results after a small number of iterations. For our experiments, we set the number of iterations to four.

In the context of deep neural networks, the Snake Head can also be regarded as a recurrent neural network. The locations of the vertices then represent the hidden state of the network, which is updated throughout the iteration steps until the final output is achieved.

C. Loss Function

The loss function is a crucial element of any deep learning model, as it measures how well the model is performing

and gives feedback for improving the network via backpropagation. When predicting contours, the loss function should therefore measure the similarity between the predicted contour p, represented by vertices p_i with $1 \leq i \leq V$, and the true contour t, given by the vertices t_j with $1 \leq j \leq V$.

Common loss functions for polygon regression are based on the L_1 and L_2 losses, which, following the above notation, are defined as follows:

$$L_1(p, t) = \frac{1}{V} \sum_i \| p_i - t_i \| \qquad (1)$$

$$L_2(p, t) = \frac{1}{V} \sum_i \| p_i - t_i \|^2. \qquad (2)$$

These loss functions have a fundamental issue when predicting glacier front lines. By only computing the distance between the vertices of p and t at the same index, they tacitly assume that each predicted vertex corresponds to exactly one vertex in the true contour. However, the model has no way of knowing how the ground truth vertices were placed along the true contour. In the setting of Deep Snake and DANCE, the vertices were placed equidistantly along the contour of objects that were largely convex, so this assumption did not have much negative impact there.

When predicting glacier frontlines, however, this issue becomes much more prominent due to the irregular and jagged shape of these contours. As the model essentially tries to minimize the L_1 or L_2 loss for a number of possible parameterizations of the true contour at the same time, the resulting predictions lack sharp edges and instead follow a smoothed version of the true outline.

Naturally, contour prediction is not the first task to face challenges like these. For example, in the context of time-series analysis, slight variations in timing are often less important than the general shape of the time-series. *Dynamic time warping* (DTW) is a method that was proposed by Sakoe and Chiba [29] in order to address this very issue. Given two sequences, they not only compare the pairwise differences but instead, first, find an optimal alignment between the two sequences and then calculate the distances based on that alignment.

In our setting, the parameterization of a contour takes the role of time in the original DTW. Formally, we define the DTW loss for two contours p and t to be

$$\mathcal{L}_{\text{DTW}}(p, t) = \min_{(i_k, j_k)_{k \in [K]} \in \mathcal{K}} \sum_k \| p_{i_k} - t_{j_k} \|_2^2 \qquad (3)$$

where $\mathcal{K}$ denotes the set of all possible realignments $(i_k, j_k)_{k \in [K]}$ that satisfy the following three conditions.

1) For any $i \in \{1, \ldots, V\}$, there is a k with $i_k = i$.
2) For any $j \in \{1, \ldots, V\}$, there is a k with $j_k = j$.
3) The sequences i_k and j_k are nondecreasing in k.

Under these conditions, the DTW loss can be efficiently calculated using dynamic programming [29].

A possible issue with the use of DTW as a loss function in deep learning is the fact that it is not smooth due to the minimum operator applied in (3). Seeing this, Cuturi and Blondel [30] replace the minimum with a *soft minimum* which

they define as

$$\text{softmin}_\gamma (x_1, \ldots, x_n) = -\gamma \log \sum_{k=1}^n \exp \frac{-x_k}{\gamma} \qquad (4)$$

with a smoothness parameter $\gamma > 0$. In the limit $\gamma \to 0$, the conventional minimum operator is recovered.

D. Implementation Details

A central issue with naively backpropagating the loss through the snake iteration is the fact that the early iterations show poor convergence to the target contour. This is easily fixed by stopping the gradient from flowing through the coordinates at the beginning of each snake step. To still encourage quick convergence of the contours during inference, we leverage deep supervision by including an additional loss term for each intermediate step. During training, the current contour is compared to the ground truth after each snake step, and the resulting loss is added toward the final loss for the gradient calculation. Unless otherwise stated, all models use a contour parametrization by 64 vertices.

All models in the study were trained for 500 epochs on the training dataset. We used the Adam optimizer [31] with an initial learning rate of 10^{-3} decaying to $4 \cdot 10^{-5}$ on a cosine decay schedule [32].

Our models are implemented in JAX [33] using the Haiku framework [34]. The training was conducted on a single NVIDIA RTX 3090 GPU with 24 GB of VRAM. Training an instance of the model, took around 25 h, and had an estimated energy consumption of 8.1 kWh.

IV. EXPERIMENTS AND RESULTS

A. Datasets

In order to thoroughly evaluate our model and compare it with other approaches, we choose two large-scale datasets of marine-terminating glaciers in Greenland for training and evaluation purposes, namely the *CALFIN* dataset [4] and the calving front dataset from *TU Dresden* (TUD) [13]. Both of these datasets include respective testing data. Furthermore, the Baumhoer dataset [11] consists of synthetic aperture radar (SAR) data of Antarctic glaciers, thus serving as a benchmark for the models' ability to generalize to a different data modality and a different ice sheet.

1) CALFIN Dataset: The CALFIN dataset consists of near-infrared data from the various Landsat missions and is most notable for the long time span of acquisition times, ranging from 1972 to 2019. Its spatial coverage is 66 Greenlandic glaciers, which amount to 1541 Landsat scenes. In an effort to improve generalizability to different sensors, the training dataset also includes 232 single-polarization Sentinel-1 scenes of glaciers in Antarctica. The corresponding test dataset consists of 162 Landsat near-infrared scenes. For all of the mentioned scenes, the calving fronts were manually delineated.

2) TUD Dataset: In contrast to this, the TUD dataset puts its focus on the eighth iteration of the Landsat mission, providing a dense time-series of recent acquisitions of Greenland's marine-terminating glaciers. The captured scenes

range from 2013 to 2021 for a total of 1127 tiles. For studies related to feature importance and data fusion, it includes the full multispectral imagery, as well as topography data and texture information derived using gray-level co-occurrence matrix statistics. For interoperability with the other datasets, only the panchromatic imagery is used in this study.

3) Baumhoer Dataset: Another test set for evaluating the generalization of the trained models is given by the Baumhoer dataset [11]. This dataset is vastly different from the other datasets, as the imagery is not from Greenland, but from Antarctica instead. Furthermore, it consists of Sentinel-1 SAR imagery, which marks a second challenge in generalization. While the original dataset is not openly available, an evaluation subset is distributed along with the CALFIN dataset [4]. In order to keep this study fully reproducible, we only use this publicly available subset of the Baumhoer dataset. The used testing set consists of 62 Sentinel-1 scenes of glaciers in Antarctica from the year 2018.

B. Evaluation Metric

As there is no uniquely defined distance metric between two curves, many different metrics are being used for evaluating the accuracy of predicted glacier frontline positions [8]. In our work, we adapt the Polis metric [35], which was originally proposed for measuring the dissimilarity between building footprints. For two polylines v and w, with I and J vertices, respectively, it is defined as the average distance of any vertex to the respective other polyline

$$p(v, w) = \frac{1}{I} \sum_{i=1}^{I} d(v_i, w) + \frac{1}{J} \sum_{j=1}^{J} d(w_j, v)$$

where $d(v_i, w)$ denotes the distance between vertex v_i and the closest point on the polyline w. Note that this closest point w does not need to be a vertex, but may be a point between vertices as well.

Compared to other existing metrics, like the Fréchet distance [36], which is defined as the solution to a min-max problem, the Polis metric is more easily interpretable as the "average" distance between the two curves. Furthermore, it was chosen due to its symmetry and the fact that it takes all predicted points into consideration.

C. Comparison With Other Models

For our comparison study, we train a number of different models in order to compare their performance on the test datasets. To compare with the state of the art in calving front detection and contour-based outline detection, we include both pixel-wise and contour-based models.

1) Pixel-Wise Models: This first group of methods consists of pixel-wise segmentation models that are known to work well for calving front detection.

a) UNet [9]: This model is a popular semantic segmentation model that serves as a strong baseline for many segmentation tasks. It has been successfully applied to calving front detection [11].

b) DeepUNet [13]: The model developed and used by Loebel et al. [13]. Its main difference from the original UNet model is the addition of two down- and upsampling steps, which make the model deeper and more aware of spatial context.

c) HED-UNet [5]: A combination of the UNet model with an edge detection model, HED-UNet was originally developed to detect glacier frontlines on the Baumhoer dataset [11].

d) CALFIN [4]: This model was introduced by Cheng et al. [4]. The model is based on the segmentation architecture DeepLabv3+ [15]. It is the first to leverage the potential of the Xception network for calving front detection.

2) Contour-Based Models: For a comparison to existing contour-based models, we also include models from this group in the comparison. It should be noted that unlike the pixel-wise models above, they were not developed for calving front detection.

a) Deep AC [22]: One of the first works to combine AC models with deep learning, this model uses a 2-D CNN to predict an offset field that points toward the nearest contour point from each pixel and then evolves a contour along this offset field.

b) Deep Snake [23]: Originally proposed as a contour-based model for instance segmentation, we have made slight changes to this model to perform calving front detection. Specifically, the circular convolutions in the network were replaced with regular 1-D convolutions, as the predicted contours for calving fronts should be open polylines and not closed polygons. Furthermore, the object detection head of the model was removed as with calving fronts, there is always exactly one contour to be predicted. For a fair comparison, we train and evaluate this model not only with the ResNet-50 backbone but also with the Xception backbone.

c) DANCE [24]: An iteration of the Deep Snake [23] model, DANCE introduces an edge attention map that speeds up the evolution for vertices far from the true edge and slows the evolution for vertices on the true edge. We applied the same adaptations to this model as to the Deep Snake model.

As CALFIN provides the largest and longest record of glacier observation, we train all models on the CALFIN training set and then evaluate them on the CALFIN, TUD, and Baumhoer test sets. The numerical evaluation results for this comparison study are displayed in Table I, and some visual results are displayed in Fig. 4. For the proposed Charting Outlines by Recurrent Adaptation (COBRA) model, we train three randomly initialized models and report mean and standard deviation across these three runs.

Comparing the pixel-wise models, we can reproduce the increased performance of the calving front-specific models over the baseline UNet. All three of these models, namely DeepUNet, HED-UNet, and the CALFIN cut down the average prediction error from UNet's 224 m to the range of 130–138 m on the CALFIN test set. There is, however, a difference in generalization to the other datasets, where the DeepUNet seems to generalize best to SAR imagery, and the CALFIN generalizes better than others on the TUD dataset, which is also based on Landsat imagery.
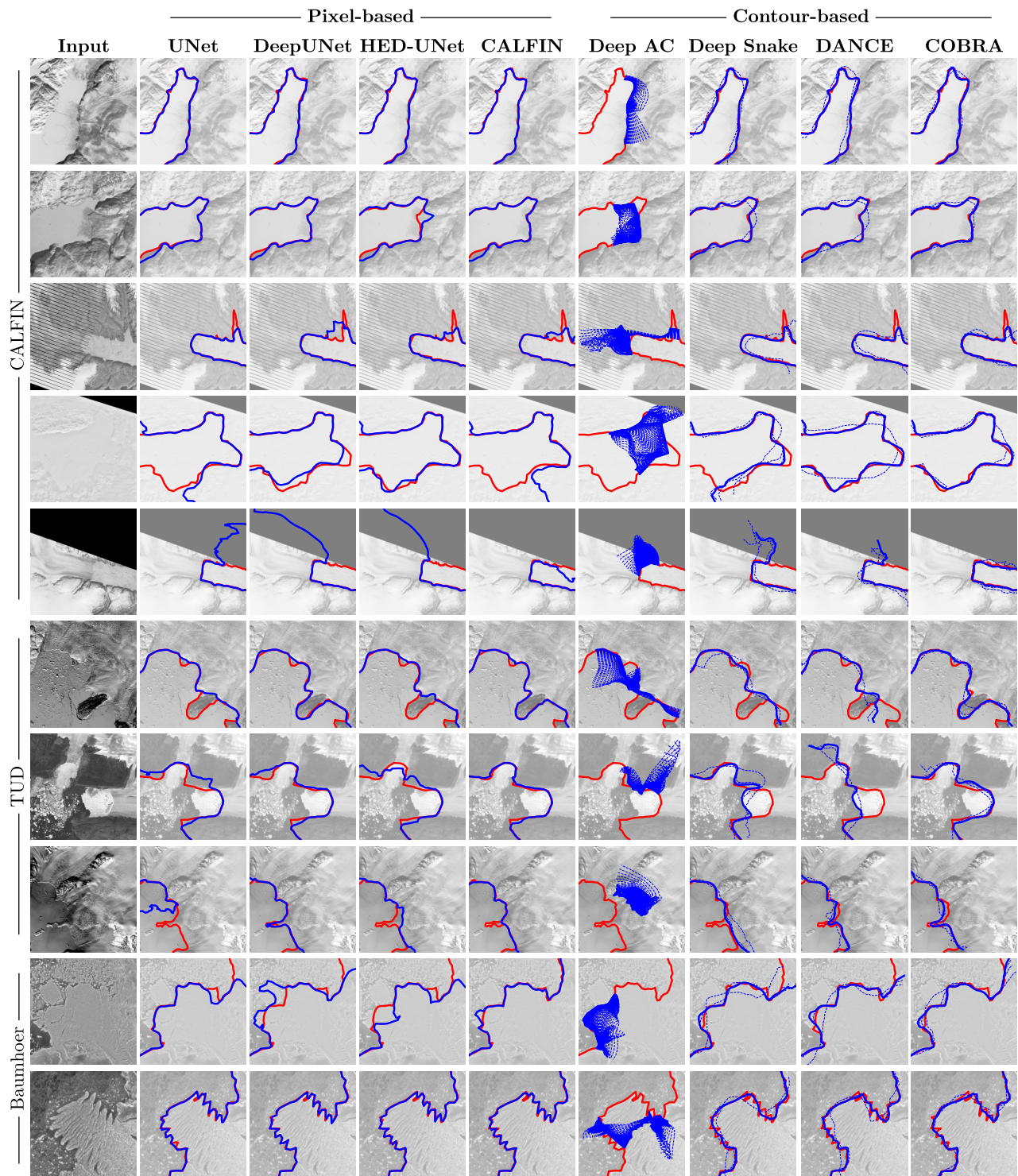
Fig. 4. Visualization of prediction results (blue) for the different models and corresponding ground truth (red) on the test datasets. For the iterative, contour-based models intermediate results are displayed as blue dashed lines. Best viewed in color.

Looking at the contour-based models, the Deep AC model falls behind the competition and performs the worst out of all the models in our experiments. The bad performance of the Deep AC model is easily explained when looking at the visual results in Fig. 4. It stems from the fact that its predictions tend to only represent a part of the desired curve, as there is no regularization term that forces the prediction to cover the entire calving front.

With the exception of the Deep AC model [22], the contour-based approaches perform considerably better on the CALFIN evaluation than their pixel-based counterparts, especially when using the Xception backbone network.

On the other hand, generalizing to the Antarctic Baumhoer dataset is particularly hard for contour-based methods. We attribute this to the presence of jagged floating ice-tongues (see Fig. 4, last row), which are not observed in the same

TABLE I

MEAN DEVIATION (POLIS METRIC) OF THE TRAINED MODELS ON THE EVALUATED TEST SETS

	CALFIN	TUD	Baumhoer
UNet [9]	224 m	288 m	122 m
DeepUNet [13]	138 m	241 m	92 m
HED-UNet [5]	130 m	231 m	122 m
CALFIN [4]	138 m	159 m	99 m
Deep Active Contours [22]	515 m	699 m	652 m
Deep Snake RN50 [23]	289 m	467 m	247 m
Deep Snake Xception [23]	123 m	316 m	130 m
DANCE RN50 [24]	118 m	290 m	131 m
DANCE Xception [24]	103 m	272 m	102 m
COBRA (mean of 3)	99 m	144 m	99 m
COBRA (standard deviation)	±10 m	± 21m	±12m

TABLE II

UNCERTAINTY QUANTIFICATION: PEARSON CORRELATION BETWEEN MODEL UNCERTAINTY AND ACTUAL PREDICTION ERROR (POLIS METRIC)

Dataset	CALFIN	TUD	Baumhoer
UNet [9]	0.3603	0.3864	0.1873
DeepUNet [13]	0.2762	0.3726	0.4191
HED-UNet [5]	0.3391	0.5518	0.3989
CALFIN [4]	0.1790	0.3449	0.2337
Deep Active Contours [22]	0.0842	0.2190	0.1759
Deep Snake Xception [23]	0.2009	0.4959	0.4508
DANCE Xception [24]	0.2972	0.3346	0.3589
COBRA	0.4811	0.4414	0.6031

way during training. Such features lead to more complex outlines, which need more vertices for their representation, as can be seen from the results of the study on vertex numbers in Section IV-E2.

Overall, our proposed network outperforms both the pixel-based and other contour-based models by a considerable margin on the CALFIN and TUD evaluations. Even when generalizing to the radically different Baumhoer dataset, our model still maintains a respectable performance.

Inference results from the COBRA model for the three testing datasets are available for online viewing and as a shapefile download at https://github.com/khdlr/COBRA/tree/master/inference_results.

D. Quantifying Uncertainty With Contour Models

With deep learning models growing ever more complex, quantifying the uncertainty of their predictions at inference time has become an important consideration when working with such models. Deep learning models being over-confident in their predictions is a common issue [37]. As the models are usually trained on definite ground truth, the models are never taught to concede their uncertainty in ambiguous cases.

One elegant method for the quantification of network uncertainties is known as MC dropout (MCD). In their seminal study, Gal and Ghahramani [7] demonstrated that a deep learning model trained with dropout layers can be interpreted as approximated Bayesian inference in a deep Gaussian process. Samples from the posterior distribution approximated by such a model can be recovered quite easily by enabling the dropout layers not only at training time but also at inference time. It has been shown that MCD can quantify model uncertainties well for remote sensing tasks like aerial image segmentation [38]. Recently, Hartmann et al. [39] also successfully applied a Bayesian UNet for the segmentation of glaciers in SAR imagery.

As the MCD method is simple to implement and evaluate compared to other uncertainty quantification methods, we choose this approach for quantifying uncertainties in the model predictions. In order to estimate the hardness of samples at inference time and get an estimate for the model uncertainty, we calculate the original, deterministic model prediction, as well as multiple additional predictions using

the MCD technique. If these predictions all line up well with the original model prediction, the model can be assumed to be quite certain of its prediction. On the other hand, a large deviation between the original model prediction and the MCD predictions corresponds to ambiguity in the model output, implying a potentially higher prediction error.

Taking the ten MC samples and the model's deterministic prediction, we estimate the model uncertainty as the average Polis-distance of each MC sample from the deterministic prediction.

Our hypothesis is that the explicit edge parameterization by vertices lends itself much better to uncertainty quantification from these posterior samples than dense pixel-wise predictions, due to the fact that the explicit representation requires much fewer parameters. With fewer parameters, the covariance between the parameters becomes more tractable, and therefore easier to approximate for any model.

For our uncertainty quantification study, we apply MCD with a dropout rate of 20% to the aforementioned models. After training the models, we draw ten predictions with enabled dropout (posterior samples) per model for each test scene in order to assess the quality of the uncertainty quantification.

Fig. 5 shows the posterior samples obtained using the MCD models. It can be observed that the pixel-wise calving front detectors can collapse completely on hard scenes. All the evaluated pixel-wise methods suffer from this phenomenon, suggesting that it is indeed related to the mode of representation. Inspection of the underlying segmentation masks suggests that this is due to the fact that when working with segmentation masks, small changes in the segmentation can completely change the topology of the prediction as previously connected regions can become disconnected and vice-versa. Due to this effect, estimating the model uncertainty using MCD can overestimate the hardness of the samples for these models on easy scenes.

By their design, contour-based methods do not have this limitation, as they predict the frontline directly. Among these models, the DANCE architecture seems affected by similar issues as the pixel-wise models, which we attribute to the fact that DANCE incorporates an intermediate dense prediction. Overall, both Deep Snake [23] and our proposed model appear to be best suited for uncertainty quantification using MCD, with very similar samples on easy scenes, and deviating
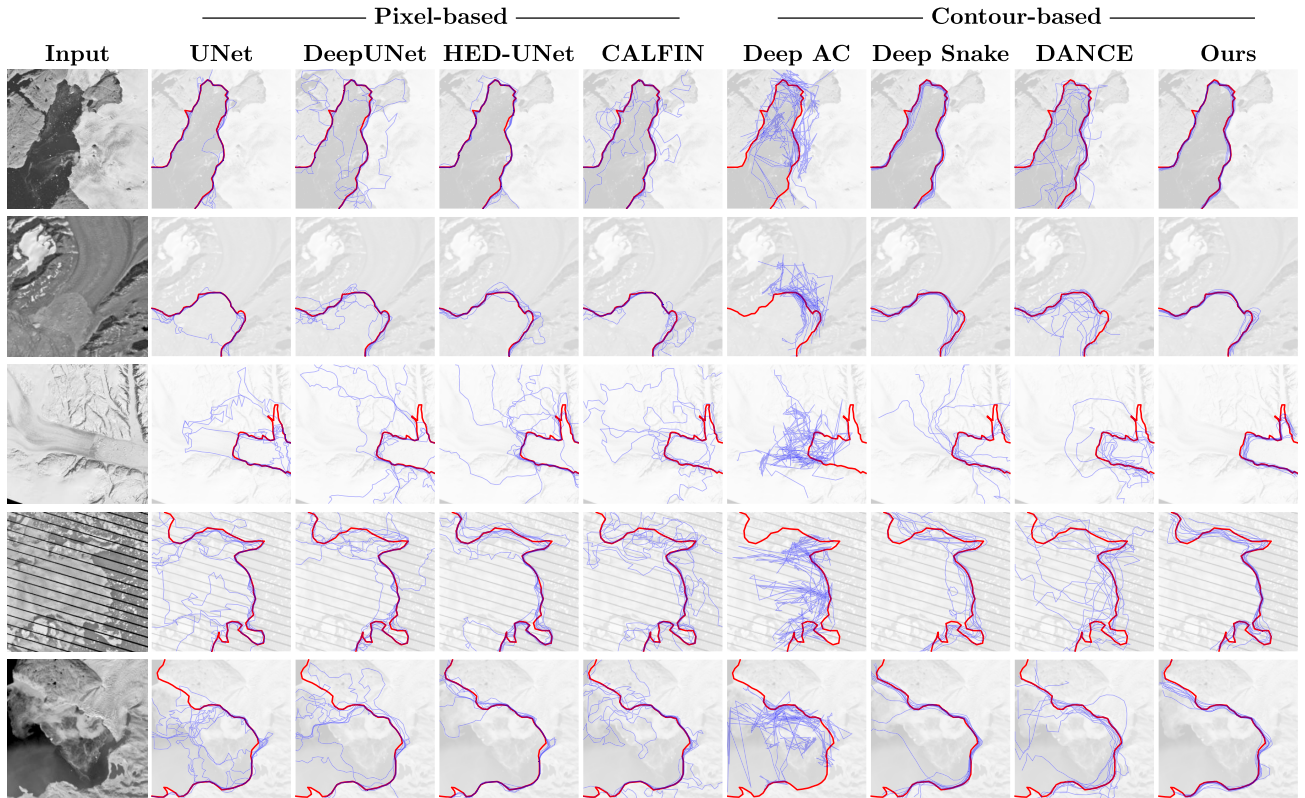
Fig. 5. Visualization of posterior samples obtained using MCD (blue) from the different models overlaid on top of the ground truth (red) for scenes from the CALFIN test set.

samples in areas that are harder to delineate. Among the pixel-based methods, HED-UNet [5] appears to be the best at quantifying its uncertainty.

In an effort to numerically evaluate the uncertainty quantification, we then calculate the Pearson correlation coefficient between the model uncertainty and the actual prediction error. A high correlation between these two variables corresponds to better uncertainty quantification, as the model should only be certain when its prediction is actually correct, while a high model uncertainty should be indicative of the prediction possibly being far from the ground truth.

The results of this evaluation are displayed in Table II. Our model is leading the evaluation for the CALFIN and Baumhoer datasets, reaching respectable Pearson coefficients of 0.4811 and 0.6031, respectively. On the TUD dataset, HED-UNet and the classic Deep Snake outperform our model on the uncertainty benchmark, achieving Pearson coefficients of 0.5518 and 0.4959. Still, our model scores decently with a Pearson coefficient of 0.4414.

These observations support our hypothesis that the contour representation is better suited for uncertainty quantification.

E. Ablation Studies

In order to better understand how the design decisions help our proposed model to improve upon the existing contour-based methods, we conduct a number of ablation studies to quantify the value of the network's components.

1) Loss Function: The rationale for implementing Soft-DTW loss for our model was the assumption that the model cannot always correctly guess the placement of the vertices

TABLE III

RESULTS OF THE LOSS FUNCTIONS ABLATION STUDY. DEVIATIONS CALCULATED USING THE POLIS METRIC

Loss Function	CALFIN	TUD	Baumhoer
L_2	114 m	312 m	119 m
L_1	102 m	296 m	111 m
DTW	90 m	236 m	91 m
SoftDTW	99 m	144 m	99 m

TABLE IV

RESULTS OF THE STUDY ON THE NUMBER OF VERTICES. DEVIATIONS CALCULATED USING THE POLIS METRIC

Vertices	16	32	64	128	256
CALFIN	120 m	98 m	99 m	98 m	116 m
TUD	252 m	209 m	144 m	214 m	261 m
Baumhoer	128 m	102 m	99 m	85 m	128 m

along the ground truth contour. Indeed, we observe better performance when using a time-warping loss, as can be seen in Table III. Surprisingly, the difference between DTW and its smooth SoftDTW variant is rather small, which suggests that the theoretical advantage of SoftDTW's smoothness does not matter much in practice for this application.

2) Number of Vertices: When choosing the number of vertices to represent the contours, a balance needs to be taken between too few vertices and too many vertices. Too low a number of vertices will not allow the model to sufficiently approximate the true contour, while too many vertices should lead to overfitting and issues in communicating information

TABLE V

RESULTS OF THE STUDY ON THE NUMBER OF ITERATIONS. DEVIATIONS CALCULATED USING THE POLIS METRIC

Iterations	2	3	4	5	6	7
CALFIN	193 m	147 m	99 m	157 m	152 m	157 m
TUD	318 m	262 m	144 m	271 m	267 m	278 m
Baumhoer	148 m	119 m	99 m	110 m	111 m	106 m

TABLE VI

RESULTS OF THE BINARY ABLATION STUDY. DEVIATIONS CALCULATED USING THE POLIS METRIC

Ablation	CALFIN	TUD	Baumhoer
Full Model	99 m	144 m	99 m
No Coordinate Features	95 m	301 m	100 m
No Gradient Stopping	397 m	285 m	531 m
No Deep Supervision	102 m	207 m	118 m
No Shared Weights	88 m	294 m	141 m

between vertices far apart in the sequence. In order to experimentally find a good setting for the number of vertices, we train COBRA configurations with different numbers of vertices. For computational efficiency, we always set the number of vertices as a power of two, choosing 16, 32, 64, 128, and 256 as possible vertex counts. The results of these experiments are displayed in Table IV.

For all three datasets, we observe that with increasing vertex count, performance decreases toward both ends of the tested range, which suggests that there is indeed a sweet spot around the middle of the evaluated range. For the CALFIN dataset, there appears to be an optimal performance plateau from 32 to 128 vertices while for the TUD dataset, 64 vertices are optimal. Interestingly the Baumhoer dataset seems to require a higher number of vertices for the best performance, reaching the best performance at 128 vertices. We attribute this to the aforementioned higher complexity of calving fronts in Antarctica. In practice, we recommend choosing the number of vertices accordingly to the complexity of the calving fronts of the region of interest. In general, setting it to 64 offers overall good performance across all study areas concerned in this study, the selection of which could be quite representative for large-scale applications.

3) Number of Iterations: A fundamental hyperparameter of our network is the number of iterations of the Snake Head. When given too few iterations, the model will likely not have enough capacity to converge to the right contour. On the other hand, given a large number of iterations, we expect the model to overfit the training set and generalize worse to unseen scenes. In order to find evidence for these hypotheses, we conduct a study on the number of iterations where we retrain COBRA models with iteration numbers from two to seven. The results in Table V suggest that there is indeed a sweet spot at four iterations. Starting from two iterations, performance improves considerably on all evaluation datasets up until four iterations. After that, increasing the number of iterations decreases the performance again. Therefore, we set the number of iterations for our model to four.

4) Coordinate Features: Including the vertex coordinates as additional features allows the Snake Head to take the distance and relative position of the vertices into account, but could also introduce a source of overfitting. In the ablation study (see Table VI), we observe that these coordinate features improve performance slightly on the CALFIN test set and drastically improve performance on the TUD dataset, where the average deviation is more than halved. For the Baumhoer dataset, performance degrades slightly when including coordinate features. This suggests that the coordinate features help the model to learn implicit shape priors for Greenlandic glacier calving

fronts, which are not helpful when transferring the model to the Antarctic calving fronts in the Baumhoer dataset.

5) Gradient Stopping: Originally, the idea of stopping the gradients from flowing through the vertex coordinates between iterations was introduced to improve the convergence of the model. However, the "No Gradient Stopping" ablation in Table VI shows that this choice is essential for the performance of the model. Without gradient stopping, the model predictions deteriorate to a degree where they are worse than the predictions of the baseline UNet model. We attribute this to numerical instabilities in the texture sampling procedure that is used to translate between the feature maps and vertex features, which can arise from letting gradients flow through the vertex positions.

6) Deep Supervision: During training, we calculate a loss term after each snake iteration and sum up these individual loss terms for the final loss. To quantify the contribution of this deep supervision, we also evaluate a model trained without intermediate loss terms, displayed as "No Deep Supervision" in Table VI. While the in-distribution samples from the CALFIN test set do not improve much with deep supervision, generalization on TUD and Baumhoer is improved by this change.

7) Weight Sharing: The underlying hypothesis for shared weights in the Snake Head iterations was the assumption that a single set of weights would lead to better generalization results than applying a series of multiple distinct Snake Heads. The ablation results for "No Shared Weights" in Table VI support this hypothesis. On the CALFIN test set, the prediction accuracy is nearly constant between the model with shared weights and the one with distinct weights. However, on the other test sets, the performance improves considerably when sharing the weights between the Snake Head iterations.

V. CONCLUSION

We proposed an approach to detecting calving fronts that directly predict the desired contours instead of predicting dense masks as an intermediate output. By training our method and existing methods on the CALFIN dataset, we showed that this new approach outperforms previous methods both on the CALFIN and TUD test sets, and exhibits competitive performance on the Baumhoer test set. In our ablation study, we showed the importance of network elements like the loss function, stopping gradient flow in the Snake Head, and sharing the weights between iterations.

Furthermore, we showed that deep AC models not only provide accurate delineations of calving fronts but also are naturally suited for the quantification of the prediction uncertainties.

We hope that this study can inspire new ways of approaching similar tasks in remote sensing where boundaries are studied, like grounding line detection or firn line detection.

Finally, we believe that the shift in representation from pixel-wise masks to GIS-native data structures like polylines will not only reduce the computational burden but also allow for exciting new approaches like enforcing physical constraints and temporal consistency or analysis across different coordinate reference systems.

REFERENCES

[1] J. Mouginot et al., "Forty-six years of Greenland ice sheet mass balance from 1972 to 2018," *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 19, pp. 9239–9244, May 2019.

[2] M. Tedesco, S. Doherty, X. Fettweis, P. Alexander, J. Jeyaratnam, and J. Stroeve, "The darkening of the Greenland ice sheet: Trends, drivers, and projections (1981–2100)," *Cryosphere*, vol. 10, no. 2, pp. 477–496, Mar. 2016.

[3] A. Shepherd et al., "Mass balance of the Greenland ice sheet from 1992 to 2018," *Nature*, vol. 579, no. 7798, pp. 233–239, Mar. 2020.

[4] D. Cheng et al., "Calving front machine (CALFIN): Glacial termini dataset and automated deep learning extraction method for Greenland, 1972–2019," *Cryosphere*, vol. 15, no. 3, pp. 1663–1675, 2021.

[5] K. Heidler, L. Mou, C. Baumhoer, A. Dietz, and X. X. Zhu, "HED-UNet: Combined segmentation and edge detection for monitoring the Antarctic coastline," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4300514.

[6] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, Jan. 1988.

[7] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. 33rd Int. Conf. Mach. Learn.*, M. F. Balcan and K. Q. Weinberger, Eds. New York, NY, USA, vol. 48, Jun. 2016, pp. 1050–1059.

[8] C. Baumhoer, A. Dietz, S. Dech, and C. Kuenzer, "Remote sensing of Antarctic glacier and ice-shelf front dynamics—A review," *Remote Sens.*, vol. 10, no. 9, p. 1445, Sep. 2018.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.

[10] Y. Mohajerani, M. Wood, I. Velicogna, and E. Rignot, "Detection of glacier calving margins with convolutional neural networks: A case study," *Remote Sens.*, vol. 11, no. 1, p. 74, Jan. 2019.

[11] C. A. Baumhoer, A. J. Dietz, C. Kneisel, and C. Kuenzer, "Automated extraction of Antarctic glacier and ice shelf fronts from Sentinel-1 imagery using deep learning," *Remote Sens.*, vol. 11, no. 21, p. 2529, Oct. 2019.

[12] M. Periyasamy, A. Davari, T. Seehaus, M. Braun, A. Maier, and V. Christlein, "How to get the most out of U-Net for glacier calving front segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1712–1723, 2022.

[13] E. Loebel et al., "Extracting glacier calving fronts by deep learning: The benefit of multispectral, topographic, and textural input features," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4306112.

[14] M. Holzmann, A. Davari, T. Seehaus, M. Braun, A. Maier, and V. Christlein, "Glacier calving front segmentation using attention U-Net," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2021, pp. 3483–3486.

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Computer Vision—ECCV 2018*, vol. 11211. Cham, Switzerland: Springer, 2018, pp. 833–851.

[16] E. Zhang, L. Liu, L. Huang, and K. S. Ng, "An automated, generalized, deep-learning-based method for delineating the calving fronts of Greenland glaciers from multi-sensor remote sensing imagery," *Remote Sens. Environ.*, vol. 254, Mar. 2021, Art. no. 112265.

[17] N. Gourmelon, T. Seehaus, M. Braun, A. Maier, and V. Christlein, "Calving fronts and where to find them: A benchmark dataset and methodology for automatic glacier calving front extraction from synthetic aperture radar imagery," *Earth Syst. Sci. Data*, vol. 14, no. 9, pp. 4287–4313, Sep. 2022. [Online]. Available: https://essd.copernicus.org/articles/14/4287/2022/

[18] M. Marochov, C. R. Stokes, and P. E. Carbonneau, "Image classification of marine-terminating outlet glaciers in Greenland using deep learning methods," *Cryosphere*, vol. 15, no. 11, pp. 5041–5059, Nov. 2021.

[19] J. Liu, E. M. Enderlin, H.-P. Marshall, and A. Khalil, "Automated detection of marine glacier calving fronts using the 2-D wavelet transform modulus maxima segmentation method," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9047–9056, Nov. 2021.

[20] A. Davari, C. Baller, T. Seehaus, M. Braun, A. Maier, and V. Christlein, "Pixelwise distance regression for glacier calving front detection and segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5224610.

[21] T. Klinger, M. Ziems, C. Heipke, H. W. Schenke, and N. Ott, "Antarctic coastline detection using snakes," *Photogramm. Fernerkund. Geoinf.*, vol. 2011, no. 6, pp. 421–434, Dec. 2011.

[22] C. Rupprecht, E. Huaroc, M. Baust, and N. Navab, "Deep active contours," 2016, *arXiv:1607.05074Cs*.

[23] S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, and X. Zhou, "Deep snake for real-time instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 8530–8539.

[24] Z. Liu, J. H. Liew, X. Chen, and J. Feng, "DANCE: A deep attentive contour model for efficient instance segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Waikoloa, HI, USA, Jan. 2021, pp. 345–354.

[25] M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, "The importance of skip connections in biomedical image segmentation," in *Deep Learning and Data Labeling for Medical Applications*. Cham, Switzerland: Springer, 2016, pp. 179–187.

[26] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1800–1807.

[27] A. van den Oord et al., "Parallel WaveNet: Fast high-fidelity speech synthesis," in *Proc. 35th Int. Conf. Mach. Learn. (ICML)*, J. G. Dy and A. Krause, Eds. Stockholmsmässan, Stockholm, Sweden, vol. 80, Jul. 2018, pp. 3915–3923. [Online]. Available: http://proceedings.mlr.press/v80/oord18a.html

[28] Z. Wang and Y.-J. Liu, "Active contour model by combining edge and region information discrete dynamic systems," *Adv. Mech. Eng.*, vol. 9, no. 3, 2017, Art. no. 1687814017692947.

[29] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.

[30] M. Cuturi and M. Blondel, "Soft-DTW: A differentiable loss function for time-series," in *Proc. 34th Int. Conf. Mach. Learn.*, D. Precup and Y. W. Teh, Eds. vol. 70, Aug. 2017, pp. 894–903.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

[32] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. 5th Int. Conf. Learn. Represent.*, Toulon, France, Apr. 2017, pp. 1–16.

[33] J. Bradbury et al. (2018). *JAX: Composable Transformations of Python+NumPy Programs*. [Online]. Available: http://github.com/google/jax

[34] T. Hennigan, T. Cai, T. Norman, and I. Babuschkin. (2020). *Haiku: Sonnet for JAX*. [Online]. Available: http://github.com/deepmind/dm-haiku

[35] J. Avbelj, R. Müller, and R. Bamler, "A metric for polygon comparison and building extraction evaluation," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 1, pp. 170–174, Jan. 2015.

[36] H. Alt and M. Godau, "Computing the Fréchet distance between two polygonal curves," *Int. J. Comput. Geometry Appl.*, vol. 5, nos. 1–2, pp. 75–91, Mar. 1995.

[37] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Intl. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2017, pp. 1321–1330.

[38] C. Dechesne, P. Lassalle, and S. Lefèvre, "Bayesian U-Net: Estimating uncertainty in semantic segmentation of Earth observation images," *Remote Sens.*, vol. 13, no. 19, p. 3836, Sep. 2021.

[39] A. Hartmann, A. Davari, T. Seehaus, M. Braun, A. Maier, and V. Christlein, "Bayesian U-Net for segmenting glaciers in SAR imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2021, pp. 3479–3482.

Konrad Heidler (Student Member, IEEE) received the bachelor's degree in mathematics and the master's degree in mathematics in data science from the Technical University of Munich (TUM), Munich, Germany, in 2017 and 2020, respectively, where he is currently pursuing the Ph.D. degree.

His main research interests are remote sensing, computer vision, and mathematical foundations of machine learning. His research work focuses on the application of deep learning in polar regions.

Lichao Mou received the bachelor's degree in automation from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2012, the master's degree in signal and information processing from the University of Chinese Academy of Sciences (UCAS), Beijing, China, in 2015, and the Dr.-Ing. degree from the Technical University of Munich (TUM), Munich, Germany, in 2020.

He is currently a Guest Professor with the Munich AI Future Laboratory AI4EO, TUM, and the Head of the Visual Learning and Reasoning Team, Department "EO Data Science," Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Weßling, Germany. Since 2019, he has been an AI Consultant with the Helmholtz Artificial Intelligence Cooperation Unit (HAICU), Oberschleißheim, Germany. In 2015, he spent six months at the Computer Vision Group, University of Freiburg, Freiburg im Breisgau, Germany. In 2019, he was a Visiting Researcher with the Cambridge Image Analysis Group (CIA), University of Cambridge, Cambridge, U.K. From 2019 to 2020, he was a Research Scientist with DLR-IMF.

Dr. Mou was a recipient of the First Place in the 2016 IEEE Geoscience and Remote Sensing Society (GRSS) Data Fusion Contest and the Finalist for the Best Student Paper Award at the 2017 Joint Urban Remote Sensing Event and the 2019 Joint Urban Remote Sensing Event.

Erik Loebel received the bachelor's degree in geodesy and geoinformation and the master's degree in geodesy from the Technische Universität Dresden (TU Dresden), Dresden, Germany, in 2016 and 2019, respectively, where he is currently pursuing the Ph.D. degree with the Chair of Geodetic Earth System Research.

His research interests include remote sensing and Earth observation, machine learning, signal processing, and time series analysis, with a special focus on the cryosphere and polar regions. His research aims at developing and applying deep learning methods for monitoring glacier changes in Greenland.

Mirko Scheinert received the Ph.D. degree in orbit dynamics of low-flying satellites from the University of Stuttgart, Stuttgart, Germany, in 1994.

He contributes to graduate lectures in geodesy and is active in the supervision of Ph.D. students. He is a Senior Scientist with the Chair of Geodetic Earth System Research, Technische Universität Dresden (TU Dresden), Dresden, Germany. His research focuses on the determination of solid Earth deformation by geodetic methods (GNSS and gravimetry), gravity field analysis and geoid determination, the application of airborne methods in geodesy, and geodetic Earth system research. For more than 25 years, he has been active in polar research, among others conducting in situ measurements in Antarctica, Greenland, and Patagonia.

Sébastien Lefèvre (Senior Member, IEEE) received the M.Sc. degree in computer engineering from the University of Technology of Compiègne, Compiègne, France, in 1999, the Ph.D. degree in computer science from the University of Tours, Tours, France, in 2002, and the Habilitation degree in computer science from the University of Strasbourg, Strasbourg, France, in 2009.

Since 2010, he has been a Full Professor of computer science with the Université Bretagne Sud, Vannes, France, where he is involved in numerous activities related to artificial intelligence for Earth and environment observation. He is the Chair of the GeoData Science Track, EMJMD Copernicus Master in Digital Earth; the Founder of the OBELIX Team, Institute for Research in Computer Science and Random Systems; and initiated the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD) Machine Learning for Earth Observation (MACLEAN) Workshop series on machine learning for Earth observation.

Xiao Xiang Zhu (Fellow, IEEE) received the M.Sc., Dr.-Ing., and Habilitation degrees in signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She is currently the Chair Professor for data science in Earth observation with TUM and was the founding Head of the Department "EO Data Science," Remote Sensing Technology Institute, German Aerospace Center (DLR), Weßling, Germany. Since 2019, she has been a Coordinator of the Munich Data Science Research School (www.mu-ds.de). Since 2019, she has been heading the Helmholtz Artificial Intelligence—Research Field "Aeronautics, Space and Transport." Since May 2020, she has been the Principal Investigator and the Director of the International Future AI Laboratory "AI4EO—Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond," Munich. Since October 2020, she has been serving as the Director for the Munich Data Science Institute (MDSI), TUM. She was a Guest Scientist or a Visiting Professor with the Italian National Research Council (CNR-IREA), Naples, Italy, in 2009; Fudan University, Shanghai, China, in 2014; The University of Tokyo, Tokyo, Japan, in 2015; and the University of California at Los Angeles, Los Angeles, CA, USA, in 2016. She is currently a Visiting AI Professor with the ESA's Phi-lab. Her main research interests are remote sensing and Earth observation, signal processing, machine learning, and data science, with their applications in tackling societal grand challenges, such as global urbanization, UN's sustainable development goals (SDGs), and climate change.

Dr. Zhu is a member of the Young Academy (Junge Akademie/Junges Kolleg) at the Berlin-Brandenburg Academy of Sciences and Humanities, the German National Academy of Sciences Leopoldina, and the Bavarian Academy of Sciences and Humanities. She serves on the Scientific Advisory Board of several research organizations, including the German Research Center for Geosciences (GFZ) and the Potsdam Institute for Climate Impact Research (PIK). She is also an Associate Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING and an Area Editor for Special Issues of *IEEE Signal Processing Magazine*.