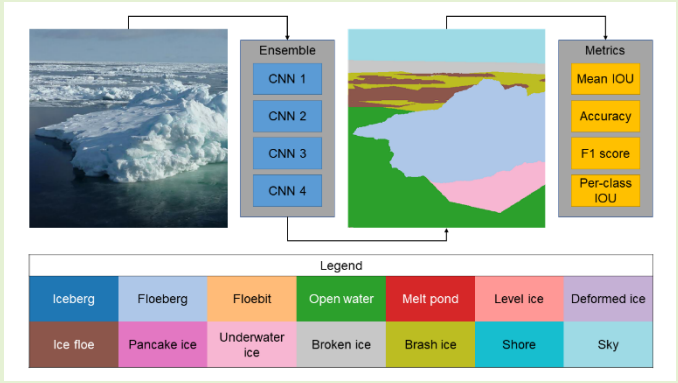
Supplementing remote sensing of ice: Deep learning-based image segmentation system for automatic detection and localization of sea ice formations from close-range optical images

Nabil Panchi, Ekaterina Kim, and Anirban Bhattacharyya

***Abstract*—This paper presents a three-stage approach for the automated analysis of close-range optical images containing ice objects. The proposed system is based on an ensemble of deep learning models and conditional random field postprocessing. The following surface ice formations were considered: Icebergs, Deformed ice, Level ice, Broken ice, Ice floes, Floe bergs, Floe bits, Pancake ice, and Brash ice. Additionally, five non-surface ice categories were considered: Sky, Open water, Shore, Underwater ice, and Melt ponds. To find input parameters for the approach, the performance of 12 different neural network architectures was explored and evaluated using a 5-fold cross-validation scheme. The best performance was achieved using an ensemble of models having pyramid pooling layers (PSPNet, PSPDenseNet, DeepLabV3+, and UPerNet) and convolutional random field postprocessing with a mean intersection over union score of 0.799, and this outperformed the best single-model approach. The results of this study show that when per-class performance was considered, the Sky was the easiest class to predict, followed by Deformed ice and Open water. Melt pond was the most challenging class to predict. Furthermore, we have extensively explored the strengths and weaknesses of our approach and, in the process, discovered the types of scenes that pose a more significant challenge to the underlying neural networks. When coupled with optical sensors and AIS, the proposed approach can serve as a supplementary source of large-scale 'ground truth' data for validation of satellite-based sea-ice products. We have provided an implementation of the approach here.**



***Index Terms*—Convolutional Neural Networks, Deep Learning, Intelligent Systems, Machine learning, Remote sensing, Sea ice, Semantic Segmentation**

I. Introduction

Accurate and reliable information about sea ice is required to promote safe and efficient maritime operations in polar regions. Many applications and scientific tasks (e.g., shipping, tourism, fishing, ocean-atmospheric heat exchange assessment, offshore activities) depend on accurate detection, monitoring, and logging of sea ice and icebergs at different spatial and temporal scales. Systematic ship-based observations of ice conditions and ice data collections are rare, and hence, a considerable amount of attention has been devoted to remote sensing of sea-ice features and icebergs from satellites and airborne platforms. The remote sensing information (e.g., ice thickness, presence of icebergs) is validated against in situ observations or similar information products. In the past few years, the number of ships equipped with optical, infrared, and thermal cameras has increased, and with that, the amount of ice information collected from ships has also increased. Optical close-range image data from ships provide ice information at very fine resolution (10 m - 1000 m); thus, they could be used to validate and supplement algorithms utilizing data from other remote sensing platforms (satellites and airborne platforms). However, optical imagery collected from ships needs to be processed before it is useful for scientific studies or maritime operations. Currently, this processing requires extensive manual intervention, limiting our ability to analyze a large

This manuscript was submitted for review on 13th December 2020.

Nabil Panchi is with the Department of Ocean Engineering and Naval Architecture, Indian Institute of Technology Kharagpur, Kharagpur West Bengal 721302, India (e-mail: panchinabil@gmail.com).

Ekaterina Kim is with the Department of Marine Technology, Norwegian University of Science and Technology, Trondheim 7491, Norway (e-mail: ekaterina.kim@ntnu.no).

Anirban Bhattacharyya is with the Department of Ocean Engineering and Naval Architecture, Indian Institute of Technology Kharagpur, Kharagpur West Bengal 721302, India (e-mail: ab@naval.iitkgp.ac.in).

number of ice images.

This paper aims to develop and validate a fast online, shipborne system that can detect ice objects and estimate their position from an optical image containing highly complex ice scenes to provide 'ground-truth' information to support satellite observations.

II. Relevant works

Remote sensing of sea-ice features from satellites has been a focus for a long time. Many algorithms have been developed for sea-ice classification over the years using remote sensing data [1]. Synthetic aperture radar (SAR) data have been used for estimation of the degree of sea ice ridging, thickness mapping of pancake ice, automatic discrimination of sea ice, determination of risk index outcome (RIO) based on International Maritime Organization's (IMO) polar code and operational sea ice charting; see, for example, [2]–[6]. Algorithms for retrieval of sea ice concentration (SIC) from remote sensing data have also been developed [7]. Microwave satellite data have been compared with many other types of satellite information of higher resolution, such as radar, visible, and infrared [8]–[10]. Comparisons of various algorithms used to calculate sea-ice parameters are presented in [11]–[14].

Considering that each data source has its strengths and sources of uncertainties, a complete examination of ice qualities can be accomplished by comparing and coordinating data from multiple data sources. Many of the aforementioned studies validate their outcomes against other forms of data. Prominent forms of data used for evaluation include direct in situ observations and manual sea-ice charts from other data sources. For example, the results from the automated algorithm proposed in [2] were compared to digitized ice charts provided by the Finnish ice service (FIS). Sea ice charts from the Arctic and Antarctic Research Institute (AARI) were used for the evaluation of sea ice types, and the corresponding operational limits were calculated in [5], [6]. Other examples of comparisons between satellite data and sea ice charts can be found in [15]–[18].

Ship-based observations also play an essential role in the validation of various remote sensing-based algorithms. A comparison of various algorithms allows for the analysis of differences between the result estimates, whereas comparison with ship data allows for the analysis of differences between the field data and results from the algorithms. It is also worth noting that the sea ice charts themselves are prepared based on conglomerations of various data sources (e.g., remote sensing data, ship-based observations, and direct in situ records). Therefore, the comparisons with sea-ice charts also indirectly include ship-based observations. Examples of validation against ship-based observations are presented in [8], [19]–[27].

Ship-based observations are irregular; however, they include ice information at a local scale (~10-1000 m). At present, most of the available ship-based observations are based upon IceWatch/ASSIST (Arctic Ship-Based Sea Ice Standardization) [28] and Antarctic Sea Ice Processes and Climate (ASPeCt) [29] protocols; see, for example, [23]–[27]. Visual observations are conducted from the ship, and the data are recorded manually in a structured form. This happens once per hour (or every three hours), and hence, ice conditions are determined at one point every 10–20 km, depending on the ship's speed. These protocols require manual interpretation of the ice scenes by a trained expert; therefore, they are limited by the rapid changes in ice conditions between observations, the subjective nature of human observers, availability and the biases of inexperienced and experienced observers, and problems associated with the visibility of ice conditions (e.g., darkness, fog). The requirement of manual interpretation also limits the number of observations collected per expedition. As an increasing number of ships start to operate in polar regions, more optical image data become available. Converting all of these raw data into useful information would require substantial manual effort, even by a trained ice expert. This motivates the development of a robust approach capable of analyzing thousands of images for the extraction of useful ice object information from close-range observations.

Previous attempts at automated analysis of ice scenes (close-range images) are either limited to a few freshwater ice features to understand/monitor river ice processes [30]–[33] or use traditional image processing techniques that are limited to broken ice with inclusions of brash, young gray, and frazil/nilas ice and specific lighting conditions [34]–[36]. A few recent studies have applied deep learning-based methods for the analysis of generalized ice scenes, but their focus is on the classification of the ice objects present in the optical image [37], [38] or segmentation of first-year ice types rather than on differentiation between ice objects (deformed ice, level ice, icebergs) [39]. Moreover, the challenges related to postprocessing ice object localizations with size-sensitive definitions, such as brash ice and ice floes, as well as the sensitivity of deep learning models to ice image distortions (e.g., grayscale and vignette effects), have never been explored before and are a subject of this paper.

Following up on our preliminary image segmentation study [40], we proposed a new approach comprising neural network ensembles and ConvCRF postprocessing. Our approach takes an optical image of an ice scene as the input and uses an ensemble of deep learning models combined with an efficient convolutional conditional random field-based postprocessing technique to automatically locate ten different ice formations (sea ice and icebergs) and four other nonice classes. Essentially, the proposed approach comprises several deep learning-based image segmentation algorithms combined with a postprocessing scheme to eliminate noise in the predictions. Deep learning (DL)-based image segmentation is not new, and several algorithms exist [41]–[45], but their performance on ice sea ice images is uncertain. To the best of our knowledge, no available approach uses an ensemble of neural networks combined with convolutional condition random field postprocessing for ice object segmentation.

To find input parameters for the approach (which modes to use in the assembly, ensemble technique, and whether to use noise removal), we have compared the capabilities of sea ice detection and ice object identification of 12 different neural networks. Fivefold cross-validation was also carried out for a

better estimation of the networks' performances. Four different model ensemble techniques were explored, and options with and without noise removal were studied. Furthermore, we have also explored and, to an extent, addressed the ice object imbalance in the training dataset for deep-learning models.

The image segmentation approach presented in this paper can be considered the base of a system that could, in the future, be mounted on the bridge of ships operating in or passing through the ice area and coupled with AIS. Eventually, we hope to reduce the requirement of interpretation of ice scenes by trained observers and increase the amount of structured ice data available for verification and supplementation of algorithms that use remote sensing data. Algorithm-based analysis of ice scenes will also ensure a more objective quantification of the sea ice parameters that could otherwise be subjective due to the difference in human experiences and opinions.

The paper is structured as follows. Section III introduces the dataset that was used to validate and evaluate the approach. Section IV presents the approach details and its input parameters, and Section V assesses the robustness of the approach through experimentation. The final sections present implications and conclusions.

III. Dataset

A new dataset had to be created for this study. The following sections briefly describe image collection, preprocessing, and the annotation process.

A. Collection

Optical images (3 channel RGB, JPEG images) of close-range ice scenes were manually collected from Google, Yandex, and Baidu search engines (338 unique images) to ensure that the images were physically plausible. Additionally, we obtained 37 unique images from the data gathered during the research cruise to the Fram Strait on the RV Lance in March 2012. Most of the collected scenes were from the Arctic region, and the remaining few were from the Antarctic region. The collected images represent images taken by different optical shipboard cameras from different angles and distances to ice objects in different weather and lighting conditions and were composed of sea ice, open water, and sky. Only a few images contained humans and animals (very small in scale compared to other features).

B. Manual Annotation

A total of 14 classes were considered for labeling. These included nine surface ice formations (Iceberg, Floeberg, Floebit, Level ice, Deformed ice, Broken ice, Ice floe, Pancake ice, and Brash ice; as defined in WMO's sea-ice nomenclature [46]) and five other nonice classes (Underwater ice, Melt pond, Open water, Sky, and Shore). The definitions of these classes are presented in TABLE I. Even though a few images contained small (in scale) humans and animals, they were ignored, and only the classes mentioned above were labeled. We used open-source image annotation software (label tool [47]) to create ground-truth labels by manually outlining the ice features. The images and their labels were verified by one ice expert, who ensured that the ice scenes were real-life-like and that ice objects were labeled correctly. For more details on the challenges and rules related to the labeling of images, we refer the reader to Table A.1 of our preliminary study [40].

C. Preprocessing

To ensure that all the images have a consistent resolution, we resized the image's shorter dimension (either height or width) to 512 pixels while maintaining the aspect ratio. Fig. 1 presents the resolutions for the original and resized images.

Fig. 2 presents the number of images and the percentage of total pixels for every class in the original and balanced datasets. It is evident from Fig. 2 that the original dataset is very imbalanced; e.g., there are 333 images containing Sky, whereas there are only 133 images containing Level ice, and only 33 images containing Melt ponds. The neural network trained directly on this dataset was biased towards the classes with a higher number of samples in the preliminary study [40]. Therefore, we oversampled the images containing minority classes (Floeberg, Floebit, Melt pond, and Pancake ice). The oversampling consisted of splitting the images into multiple parts using a sliding window of 512x512 pixels with a maximum overlap of 256 pixels between adjacent splits. This oversampled dataset was relatively better balanced than the original dataset and contained 458 images (see Fig. 2).

Before passing the images to the neural networks, we cropped the images that were larger than 512x512 pixels (to 512x512 pixels). We also applied channelwise normalization to the RGB values of each of the images. The statistics for normalization for each of the red (R), green (G), and blue (B) channels are presented in TABLE II. Training set images were cropped randomly, whereas the validation set images were cropped at the center to ensure that the comparison of validation scores between neural networks was fair. Other augmentations, such as small rotations, horizontal flips, contrast, and brightness adjustments, and perspective warping, were also applied randomly to the training set's images.

D. Dataset Splits

Considering the dataset's small size relative to other segmentation datasets [48]–[50], we chose a K-fold cross-validation scheme to evaluate the model performance. K was chosen as five because it achieves a balance between the number of images in the validation sets and the number of folds for cross-validation. The dataset was divided into six parts (folds) of nearly equal size (~77 images each). The first five folds were used for 5-fold cross-validation, and the sixth fold was set aside as a test set. For each of the validation folds (~77 images), there were ~300 training images.

It is essential to ensure that the distribution of images per class is consistent across the six folds to correctly estimate the model's performance. To ensure this consistency, we used the 'IterativeStratification' module from the scikit-multilearn library (version 0.1.0) [51], which is based upon algorithms presented in [52], [53]. Fig. 3 presents the image and pixel distributions per class. Here, the consistency in the number of images per class across the six folds is noteworthy.

Three variations of the test set were used to evaluate the generalization ability of the deep learning models. Fig. 4 presents sample images from the test set:

a) *Clear images:* Normal daylight condition (Fig. 4 (a)),

b) *Grayscale images:* To test the deep learning model's dependency on ice feature colors (Fig. 4 (b)).
c) *Images with vignette effect:* Simulating a searchlight effect at night, with only a portion of the image visible (Fig. 4 (c)).

IV. Methods

In this section, we first provide an overview of the proposed segmentation approach, which aggregates the outputs from four neural networks and applies postprocessing to obtain the final result. Then, we present the three main parts of the proposed approach: the neural network, the ensembling module, and the postprocessing module.

A. Overview

We designed a novel approach for ice object segmentation, as shown in Fig. 5. One branch passes the three-channel shipboard optical image to PSPNet, PSPDenseNet, DeepLabV3+, and UPerNet to extract preliminary outputs from the neural networks and then combines these outputs in an ensemble module. Another branch consists of a ConvCRF postprocessing module that takes the ensemble module's output and the image as the input to generate a probability map. The probability map passes through the 'argmax' layer to generate the pixel-level segmentation mask.

B. Neural Networks

The neural network architectures used for semantic segmentations have two main parts: first, the feature extractor or the backbone, which takes an image as input and progressively reduces the feature space's dimension, producing a highly nonlinear representation of the image; second, the upsampling part, also known as a decoder in some networks, which utilizes features from the feature extractor and outputs a segmentation mask.

To select models to be used in the proposed approach, we evaluated 12 neural network architectures (models), including PSPNet, PSPDenseNet, DeepLabV3+, UPerNet, DUC HDC, FCN, GCN, ENet, UNet, UResNet (UNet with ResNet backbone), SegNet, SegResNet and [45], [54]–[61]. Based on the single model performance, we settled on PSPNet, PSPDenseNet, DeepLabV3+, and UPerNet for ensembling (Fig. 5). PSPNet [54] uses a pyramid pooling module to aggregate multiscale context information in different subregions to achieve a superior result compared to simple encoder-decoder-based networks such as FCN and UNet. PSPDenseNet is based on PSPNet, but the ResNet feature extractor was replaced by a DenseNet feature extractor. DeepLabV3+ [45] combines an encoder-decoder structure with a spatial pyramid pooling module to better capture multiscale contextual information and produce sharper object boundaries by gradually recovering the spatial information. UPerNet was developed for unified perceptual parsing [55], and it integrates a feature pyramid network (FPN) [62] with a pyramid pooling module [54]. In all models except UNet, ENet, and SegNet, the encoder is either a conventional ResNet or DesnseNet architecture [43], [63]. Details of the other architectures can be found in the relevant literature [45], [54]–[61].

The neural network was trained in two stages. In the first stage, the pretrained backbone (on the ImageNet dataset [64]) was frozen, and only the upsampling part was trained. Next, the entire network was unfrozen and fine-tuned. The non-pretrained backbones (in the case of UNet, ENet, and SegNet) also remained unfrozen in the first training stage. We used the Adam optimizer [65] and the one-cycle policy [66] for both training stages. The training parameters were chosen after some preliminary investigations (presented in TABLE III).

Implementation of the training loop was derived from Fastai v1 and PyTorch 1.3.1 [67], [68]. We used the implementations of the neural network models from [69], which were implemented in PyTorch 1.3.1 [68].

C. Ensembling module

Although the quantitative performances of the four neural network architectures mentioned above are close to each other, a closer, qualitative look at the predictions revealed that each of them has its strengths and weaknesses. For example, in the same image, one ice object was predicted well by one of the networks, whereas another ice object was predicted well by another network. Furthermore, an intercomparison of per-class IOU between the models clarifies that none of the four models (PSPNet, PSPDenseNet, DeepLabV3+, UPerNet) was best at all the classes. This observation led to our hypothesis that for ice object segmentation, an ensemble of neural networks could perform better than any single neural network. In our experiments, we found ample empirical evidence to support the above hypothesis. TABLE IV provides the algorithm based on which the ensemble module works.

Although we have used the 'Mean' function to combine the model outputs, there can be several other ways to combine the neural network architectures; we also tried:

- Product ensemble – Product of the model outputs
- Max ensemble – Maximum of the model outputs

Each of these methods was implemented for both non-postprocessed and postprocessed model outputs. In the case of an ensemble of postprocessed output, Output1, Output2, Output3, and Output4 would be passed on to a ConvCRF module before the combination step (*). The mean, product, and max ensemble can be put into one category (Category I). Another way of combining the neural network output is through:

- Majority voting – Voting between all the model outputs to select one particular class, ties broken by the probability of predictions (Category II).

In this case, the model postprocessed/non-postprocessed model outputs would be passed on to the Argmax function to arrive at a segmentation mask, and then, these segmentation masks would be combined to obtain the final segmentation masks. Postprocessing cannot be applied to a majority voted result since it returns a segmentation mask of shape 512x512x1, and ConvCRF postprocessing requires probabilities as the input (shape: 512x512x14). Fig. 6 presents the schematics of the two categories of ensemble modules.

D. Postprocessing

Convolutional neural networks are powerful feature extractors; however, they do not explicitly account for the conditional dependence of pixels on each other. This introduces noise in the neural network outputs. Fig. 7 (a) presents the model outputs. The model outputs contain small, random

patches of misclassified pixels. One way of removing the noise in the predictions is by applying fully connected conditional random fields (FullCRFs) [70] on top of neural network prediction [71], [72]. However, the FullCRFs take a long time for inference. Therefore, we used a faster convolutional conditional random field (ConvCRF) [73] with default parameters as the postprocessing scheme. See Fig. 7 (b) for a postprocessed example. A quantitative analysis of postprocessed and non-postprocessed results is presented in the following section.

All the calculations in this study were performed on a virtual machine equipped with an Nvidia Tesla V100 GPU, Intel(R) Xeon(R) Gold 6148 CPU (2.40 GHz), and 32 GB of RAM. The programming language was Python (version 3.7.3).

V. Experiments

In this section, we first describe the details of the training and evaluation process for the neural networks, and then a series of ablation experiments are described. All the experiments were carried out on the dataset created in section III.

A. Evaluation

We used a 5-fold cross-validation scheme to evaluate the performances of the neural networks. To evaluate the model on each of the five validation sets, the neural network was trained on the remaining four validation sets.

An objective comparison of the segmentation models requires quantitative performance indicators. To that end, we considered Mean Intersection Over Union (Mean IOU), Accuracy, and F1 score as the performance metrics. Additionally, we explored the per-class performance of the models using per-class IOU (IOU_c). The metrics are defined as:

$$Mean\ IOU = \frac{TP}{TP+FP+FN} \tag{1}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

$$F1\ score = \frac{2TP}{2TP+FP+FN} \tag{3}$$

$$IOU_c = \frac{TP_c}{TP_c+FP_c+FN_c} \tag{4}$$

The symbols (TP, TN, FP, FN, TPc, TNc, FPc, FNc) have been defined in TABLE V.

The accuracy measure is biased towards classes that are more frequent and cover a large image area (i.e., have a high overall count of samples). Specifically, in sea-ice segmentation, classes such as Sky, Ice floe, Deformed ice, Level ice, Brash ice, and Open water, which have high overall sample counts, may sway the accuracy score, which can be problematic. Mean IOU and F1 score account for this class bias to a greater degree, and hence, a comparison involving them would provide a better picture than a comparison involving only accuracy. However, we could treat only one metric as a primary metric for decisions requiring strict objectivity (such as which model is the best). For this purpose, we have chosen the mean IOU as the primary performance metric since it is standard in the image segmentation literature [45], [74]–[77].

B. Ablation studies and discussions

1) Baseline

To establish a baseline, we cross-validated all 12 neural networks mentioned in section IV.B (Neural networks under the Methods chapter), with and without postprocessing. Fig. 8 presents the performance metrics for the 5-fold cross-validation of the deep learning models. Here, we can see that the PSPNet + postprocessing approach outperforms all other pipelines with a mean IOU score of 0.773. If we look at the top 4 single-model pipelines (i.e., pipelines with models PSPNet, PSPDenseNet, DeepLabV3+, UperNet), two points are noteworthy. First, all of them perform very well, with mean IOU and accuracy close to 0.8 and F1 scores close to 0.6. Second, their performance is consistent across the five validation folds, with a 2-3% performance increase due to postprocessing. In contrast, the rest of the models' performances vary too much across validation folds, indicating a lack of generalization.

Fig. 9 (a) presents the per-class IOU for the top four single-model pipelines. All the pipelines perform well on the classes from the left (Sky – Pancake ice, Iceberg), decently on Broken ice, Underwater ice, and Shore, and poorly on the last three classes on the right (Floeberg, Floebit, and melt pond). The per-class IOU for these three classes is very low, i.e., in the range of 0.3-0.4 for Floebit and Floeberg, whereas it was nearly 0 for Melt ponds. When analyzing the variation in performances over five validation folds, it is evident that most of the classes on the left of the plot (Sky - Underwater Ice) are more consistent, whereas others have relatively varied performances. This performance disparity is especially prominent for Floebit and Floeberg, for which the per-class IOU ranges from 0% to above 50%, depending upon the validation set. Such behavior could result from the combination of limited validation images for these two classes and a bias towards images with particular cooccurrence of ice features (e.g., images containing only level ice and deformed ice, images containing only Pancake ice and open water). We found the per-class IOU for other models (besides the top four) to be very mediocre. Furthermore, the performances of ENet and SegNet were very unusual. These two models could predict a few classes, such as sky and deformed ice, quite well, but their performances on other classes were nonexistent.

Although a qualitative comparison of the neural network architectures was not the primary focus of the study, we did look at the specific architectures of the neural networks, based on which we found that the networks with pyramid pooling modules or some modified versions of it (PSPNet, PSPDenseNet, DeepLabV3+, and UperNet) seem to perform better than other network architectures. To better understand the pyramid pooling module's effect on the model performance, we trained and cross-validated (over 5 folds) a fully convolutional neural network with a ResNet 152 (FCResNet) backbone without any pyramid pooling modules. A similar FCN with the ResNet50 backbone was used as a baseline in paper [1], proposing PSPNet. We found that the FCResNet-based segmentation approach had nearly the same mean IOU value as that of UResNet, which does not have any pyramid pooling module (mean IOU values of 0.685 and 0.675, respectively), whereas PSPNet (with the pyramid pooling module) outperformed both FCResNet and UResNet with a mean IOU

value of 0.774.

Based on the above results, PSPNet, PSPDenseNet, DeepLabV3+, and UPerNet were selected for the ensembling approach (shown in Fig. 5), and the PSPNet + ConvCRF postprocessing approach was selected as the baseline.

2) *Ablation for Ensemble approaches*

To evaluate the performance of the proposed approach, we conducted experiments with different output aggregation methods. Fig. 10 presents the performance metrics from the 5-fold cross-validation for eight different ensembles and the top four single-model approaches. As expected, most (5/8) of the ensembles perform better than the baseline. The mean ensemble shows an increase of ~3% in both the mean IOU and accuracy scores compared to the baseline. The top 3 ensembles (Product, Mean, and ConvCRF-Mean), combined with postprocessing, have near-identical performance (mean IOU scores of 0.799, 0.797, and 0.794, respectively). However, the overall performance of the mean ensemble is better than that of the product ensemble. Even though the product ensemble has a slightly higher (0.2%) mean IOU score, the accuracy and F1 score of the mean ensemble are higher by over 1% each.

The case for using the mean ensemble over the product ensemble becomes even more convincing if we consider the per-class IOU (Fig. 11 (a)). The mean ensemble improves over the per-class IOU of the baseline for 12/14 classes, whereas the product ensemble improves for 8/14 classes. The most notable increase in per-class IOU was for Pancake ice and Shore (~6% and ~4%, respectively). Unfortunately, we did not find an increase in the IOU of Melt ponds. Keeping in mind the mean ensemble's better per-class performance, we used it for evaluation on the test set.

3) *Ablation for input image distortion*

To study the effectiveness of the proposed approach on input image types that it has never seen before, we evaluated it using the three test sets created in section III.D. Fig. 12 presents the performance of the proposed approach (mean ensemble of four models + ConvCRF) on various test sets and the average values for the 5-fold cross-validation. Our approach yields equally good results on the clear test set compared to the 5-fold cross-validation (mean IOU 0.793, F1 score 0.59, and mean IOU 0.797, F1 score 0.57, respectively). When comparing the effect of distortions of the input images, grayscale images seem to have only a minor effect on the performance. In contrast, images with a vignette effect led to a more significant decline in performance, especially when considering accuracy and F1 scores. Few qualitative results from the test set are presented in Fig. 15 in the appendix.

Fig. 13 presents the per-class performance of the proposed approach on various test sets and the average values for the 5-fold cross-validation. The performance on the clear test set matches the performance from the cross-validation. The only exception to this is Floeberg, for which the per-class IOU is zero for all the test sets. A look at the test set's qualitative results revealed that the model predicted Floeberg as Deformed ice. We suspect that this is due to the similarity in the appearances of Floeberg and Deformed ice.

Based on the empirical evidence from the experiment discussed above, we found the proposed approach to be resilient to distortions (grayscale and vignette) in the input images. Grayscale image distortion only significantly affects underwater ice (decline in per-class IOU by over 50%). This response indicates that the differentiating feature for underwater ice was its specific color. In the case of images with a vignette effect, the per-class IOUs are very low for three classes: Broken Ice, Shore, and Floebit. However, this is expected, as the visibility of pixels belonging to these classes is severely affected since they tend to lie farther away from the point of image capture, making them barely visible.

Even though we did not train our models on distorted input images (grayscale images and images with vignette effects, Section III.D), there was only a minor performance deterioration for the test sets with image distortion. This shows that neural networks can generalize incredibly well. Specifically, the grayscale test set's performance indicates that when barring underwater ice, the proposed approach was indifferent to the colors of ice objects. Furthermore, the performance on the test set with the vignette effect, which realistically mimics observational conditions during night time, suggests that our approach could be extended to the segmentation of optical imagery captured during polar nights, provided that ISO does not change.

4) *Difficult classes*

All considered approaches struggled for the minority classes Floeberg, Floebit, and Melt pond. Floebit and Floeberg have a per-class IOU of ~40%, whereas Melt ponds were never predicted right. The proposed ensemble approach also did not improve the per-class IOU of these classes. One of the reasons for the poor performance for these classes could be their lack of representation in the dataset compared to other classes (refer to Fig. 9 (c), Fig. 11 (c), and Fig. 13 (c) for details).

TABLE VI presents the confusion statistics for the challenging classes. It is evident that in most cases (7/8), a large number of false negatives of Floeberg were false positives of Deformed ice. This is understandable, as floebergs have elements of deformed ice, making them both similar in appearance. Floebits have a better overall performance but a more inconsistent pattern in regard to confusion. Floebits are predicted to be icebergs, brash ice, and, most prominently, ice floes, based on whichever feature is closest in appearance.

We argue that melt ponds are inherently difficult for a deep learning model, given our setup. Investigation of the qualitative results revealed that in cases where melt ponds' appearance is bluish, it is mistaken as underwater ice; in other cases, it is either classified as open water or as a part of the ice floe. One reason for this could be that the melt ponds were too small for the neural network to be considered important, and it ended up considering it as a noisy artifact in the image. However, an investigation into the trend of per-class IOU vs. the fraction of images taken by the corresponding class revealed that classes such as Shore, Brash ice, and Broken ice have decent performance even for images in which they take up only a small portion of the image. These findings cast doubt upon the hypothesis that performance on melt ponds suffers only due to the lower percentage of images it captures. Another reason for the lower performance on melt ponds could be that since melt ponds are underrepresented in the dataset (accounting for approximately 0.2% of the total pixels), the deep-learning model does not learn enough to be able to detect and segment melt ponds accurately. We looked at two other commonly used image segmentation datasets, Cityscapes [78] and CamVid

[48], [79]. In both datasets, the category 'human' was the lowest represented category (only amounting to 1.3% and 1.2% of the total pixels, respectively). However, this is still five times higher than the percentage of pixels belonging to melt ponds in our dataset (0.2%).

Based on the findings described above, we think that there are two feasible solutions that future research can look at: first, increasing the number of images containing melt ponds in the dataset and using a specialized attention-based module in the neural network architectures [80]–[82].

5) *Ablation for ConvCRF postprocessing*

The effect of using ConvCRF postprocessing in the conducted experiments was studied, and the results showed that ConvCRF postprocessing similarly affected all our experiments. It is clear from the results that the effect of postprocessing on per-class IOU is low (but positive) for most of the classes, except for Shore, Floeberg, and Floebit. There is a loss of ~5-6% in per-class IOU for Shore and a gain of ~2-3% for Floeberg and Floebit (Refer to Fig. 9 (b), Fig. 11 (b), and Fig. 13 (b)). A qualitative examination of the model results showed that postprocessing serves its intended purpose of removing the unwanted noise from the predictions. Notwithstanding, it negatively affects classes that are present as small blobs in the image; specifically, brash ice present between two bordering ice floes, which ends up being classified as a part of one of the ice floes. Another such case involves the shore visible near the horizon; the postprocessing leads to its classification as a part of sky or level ice. However, neither of these cases is hugely concerning. The shore near the horizon is too far from the inspection point to be relevant, and the classification of a small blob of brash ice as a part of ice floe gives us a conservative estimate of the floe size.

6) *Ablation for ice and nonice objects*

All 14 classes that were considered in this study could be divided into two major categories: ice objects *(icebergs, deformed ice, level ice, broken ice, ice floes, floe bergs, floe bits, pancake ice, underwater ice, and brash ice)* and additional nonice objects *(sky, open water, shore, and melt ponds)*. The category wise mean IOU values of the proposed approach are presented in TABLE VII. The results showed that the difference between the average performance on ice objects and nonice objects ranged between 4% and 10%. The highest difference in performance was for the grayscale test set, for which the proposed approach performed ~10% better on nonice objects than on ice objects.

Although classes from both ice and nonice categories are part of an ice scene, they might have different meanings from an application point of view (i.e., evaluation of a satellite product that differentiates between ice and no ice, ships traveling in the Norwegian and the Barents Sea, which generally try to avoid all types of ice objects). Therefore, to evaluate the model's capability to differentiate between ice vs. nonice classes, we calculated the accuracy, mean IOU, and F1 scores while assuming that the proposed approach only needed to segment ice and nonice objects. These values are presented in TABLE VIII. The proposed approach was good at differentiating between ice and nonice objects in all test cases except for test images with the Vignette effect, for which the performance was relatively ~10 – 20% lower. Overall, the proposed approach performed even better on ice object vs. nonice object segmentation compared to segmentation of 14 classes by (~13% higher mean IOU for ice object vs. nonice object segmentation).

7) *Time complexity analysis*

The segmentation approach needs to be scalable, i.e., it should handle a large number of images efficiently. To that end, we analyzed the time required for our ensemble-based approach compared to the single neural network approach that we used as our baseline. In our experimental setup, the proposed ensemble approach takes 0.10 seconds to process an image end-to-end, whereas our baseline, PSPNet + ConvCRF postprocessing, takes 0.03 seconds to process the same image.

8) *Dataset and Labeling consistency*

To roughly evaluate human bias in labeling, we conducted a small experiment in which we asked five human labellers without prior ice experience to label one image. They were all provided with the same labeling software and the same set of guidelines. We then compared their labels to an expert verified label from the dataset. Fig. 14 presents all six labels (1 from the dataset + 5 from five different inexperienced labelers). We found a high overlap (avg. ~93%) between the label from the dataset and the labels from other labellers. From Fig. 14, it can be seen that the highest disagreement among the labellers was related to the boundaries of open water, underwater ice, and brash ice. Such disagreement is expected because (a) distinguishing between open water and brash ice is difficult, especially in areas with relatively low brash ice concentrations. (b) Sometimes, it is also not easy to differentiate the ice object's reflection (in this case, a floeberg) from underwater ice. Moreover, in this particular image, since the water is not very clear, it was hard to demarcate the reflection of the Floeberg and its underwater part.

The comparison between labels was limited to only one image due to its resource intensiveness. We do expect a greater disagreement among human labellers for a different ice scene. The bias in human labeling of ice objects (on close-range images) exists in the ice classification problem and has been explored in [83], [84]. The authors [83] discovered that human labellers suffer from bias towards classes that occur more frequently. Humans appear to also be biased when estimating ice concentrations in satellite images [85]. In the future, we plan to conduct visual cognition experiments with closer-range ice images to obtain a deeper understanding of human bias in the visual cognition of ice objects for comparison with computer algorithms.

Although we ensured that the image collection conditions were not extremely different across all the images in the dataset, there may be some noise because of the different cameras, angles, distances to the ice objects, etc. However, it could be argued that this noise made the training dataset more difficult than that expected in the real life, where the model will encounter images captured from a single camera at a fixed camera angle. Future direction of work should include measures to understand the impact of collection conditions on the model performance, as well as the recalibration of the models from this study for an application with a fixed camera.

VI. Conclusion

This paper's primary objective has been to develop a deep

learning-based approach that can successfully segment close-range optical ice scenes containing 14 classes (10 ice features + 4 additional classes) in view of its applicability as a source of validation data for satellite or airborne platforms. Optical images acquired during the research cruise to the Fram Strait on the RV Lance in 2012 and from various other online sources were subjected to state-of-the-art computer vision techniques. The segmentation results were extensively evaluated with respect to hand-labeled 'ground truths,' considering both overall and per-class performance. From the results of this study, we draw the following conclusions:

- The proposed approach performed very well on a significant number of the classes (Sky, Ice floe, Deformed ice, Level ice, Brash ice, Open water, and Pancake ice) in our dataset, decently on others (Broken ice, Underwater ice, Iceberg, and Shore), and poorly on three classes (Floeberg, Floebit, and Melt pond).
- The proposed ensemble approach consisting of PSPNet, PSPDenseNet, DeepLabV3+, and UPerNet + ConvCRF postprocessing significantly outperforms the PSPNet + ConvCRF baseline with gains (~2-3%, on average) in the per-class IOU compared to the best model for most of the classes (12 out of 14 classes).
- Postprocessed results using ConvCRF are quantitatively better than non-postprocessed results (~3% gain in mean IOU, on average), and overall, the mean ensemble + postprocessing emerges as the best approach with a mean IOU score of 0.799.

VII. Implications

The approach developed in this paper aims to supplement ice experts in the analysis of collected images from ships by providing segmentations of optical ice scenes. More work needs to be done to make this automated image segmentation and analysis fully operational for remote sensing applications—especially the collection and labeling of more images containing Floeberg, Floebit, and Melt ponds and introducing different labels (first-year ice, second-year ice, multiyear ice). In situ verification, validation, and possible corrections to ice segmentation would go a long way in providing more training data and improving the proposed approach.

The proposed system can provide continuous ice observations at a high temporal resolution, which can further validate/supplement satellite-derived sea ice products. Other example applications of the presented system include ice-type classification, calculation of partial and total concentrations, ice-feature size assessment using a pre-calibrated mesh as presented in Fig. 16 (Appendix), and automated conversion between optical ice scene and egg code for maritime applications. In conclusion, deep learning-based ice scene segmentation and analysis techniques applied to imagery acquired from ships operating in ice-infested waters represent an additional data source to support satellite-derived sea ice products.

Acknowledgments

The computations were performed on resources provided by UNINETT Sigma2 - the National Infrastructure for High Performance Computing and Data Storage in Norway. All the sea-ice images presented in this paper were taken from the private collection of Sveinung Løset.

References

[1] N. Zakhvatkina, V. Smirnov, and I. Bychkova, "Satellite SAR data-based sea ice classification: An overview," *Geosciences (Switzerland)*, vol. 9, no. 4. 2019, doi: 10.3390/geosciences9040152.

[2] A. Gegiuc, M. Similä, J. Karvonen, M. Lensu, M. Mäkynen, and J. Vainio, "Estimation of degree of sea ice ridging based on dual-polarized C-band SAR data," *Cryosphere*, vol. 12, no. 1, pp. 343–364, 2018, doi: 10.5194/tc-12-343-2018.

[3] P. Wadhams, G. Aulicino, F. Parmiggiani, P. O. G. Persson, and B. Holt, "Pancake Ice Thickness Mapping in the Beaufort Sea From Wave Dispersion Observed in SAR Imagery," *J. Geophys. Res. Ocean.*, vol. 123, no. 3, pp. 2213–2237, 2018, doi: 10.1002/2017JC013003.

[4] D. B. Hong and C. S. Yang, "Automatic discrimination approach of sea ice in the Arctic Ocean using Sentinel-1 Extra Wide Swath dual-polarized SAR data," *Int. J. Remote Sens.*, vol. 39, no. 13, pp. 4469–4483, 2018, doi: 10.1080/01431161.2017.1415486.

[5] M. Mäkynen *et al.*, "Satellite observations for detecting and forecasting sea-ice conditions: A summary of advances made in the SPICES Project by the EU's Horizon 2020 Programme," *Remote Sens.*, vol. 12, no. 7, 2020, doi: 10.3390/rs12071214.

[6] E. Rinne and M. Similä, "Utilisation of Cryosat-2 SAR altimeter in operational ice charting," *Cryosphere*, vol. 10, no. 1, pp. 121–131, 2016, doi: 10.5194/tc-10-121-2016.

[7] V. V. Tikhonov, M. D. Raev, E. A. Sharkov, D. A. Boyarskii, I. A. Repina, and N. Y. Komarova, "Satellite microwave radiometry of sea ice of polar regions: a review," *Izv. - Atmos. Ocean Phys.*, vol. 52, no. 9, pp. 1012–1030, 2016, doi: 10.1134/S0001433816090267.

[8] X. Pang, J. Pu, X. Zhao, Q. Ji, M. Qu, and Z. Cheng, "Comparison between AMSR2 sea ice concentration products and pseudo-ship observations of the arctic and Antarctic Sea ice edge on cloud-free days," *Remote Sens.*, vol. 10, no. 2, 2018, doi: 10.3390/rs10020317.

[9] W. N. Meier, "Comparison of passive microwave ice concentration algorithm retrievals with AVHRR imagery in arctic peripheral seas," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1324–1337, 2005, doi: 10.1109/TGRS.2005.846151.

[10] H. Wiebe, G. Heygster, and T. Markus, "Comparison of the ASI ice concentration algorithm with landsat-7 ETM+ and SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 9, pp. 3008–3015, 2009, doi: 10.1109/TGRS.2009.2026367.

[11] S. Andersen, R. Tonboe, S. Kern, and H. Schyberg, "Improved retrieval of sea ice total concentration from spaceborne passive microwave observations using numerical weather prediction model fields: An intercomparison of nine algorithms," *Remote Sens. Environ.*, vol. 104, no. 4, pp. 374–392, 2006, doi: 10.1016/j.rse.2006.05.013.

[12] N. Ivanova, O. M. Johannessen, L. T. Pedersen, and R. T. Tonboe, "Retrieval of arctic sea ice parameters by satellite passive microwave sensors: A comparison of eleven sea ice concentration algorithms," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7233–7246, 2014, doi: 10.1109/TGRS.2014.2310136.

[13] N. Ivanova *et al.*, "Inter-comparison and evaluation of sea ice algorithms: Towards further identification of challenges and optimal approach using passive microwave observations," *Cryosphere*, vol. 9, no. 5, pp. 1797–1817, 2015, doi: 10.5194/tc-9-1797-2015.

[14] E. V. Zabolotskikh, "Review of Methods to Retrieve Sea-Ice Parameters from Satellite Microwave Radiometer Data," *Izv. - Atmos. Ocean Phys.*, vol. 55, no. 1, pp. 110–128, 2019, doi: 10.1134/S0001433818060166.

[15] T. A. Agnew and S. Howell, "Comparison of digitized Canadian ice charts and passive microwave sea-ice concentrations," in *International Geoscience and Remote Sensing Symposium (IGARSS)*, 2002, vol. 1, pp. 231–233, doi: 10.1109/igarss.2002.1024996.

[16] T. Agnew and S. Howell, "The use of operational ice charts for evaluating passive microwave ice concentration data," *Atmos. - Ocean*, vol. 41, no. 4, pp. 317–331, 2003, doi: 10.3137/ao.410405.

[17] R. T. Tonboe *et al.*, "The EUMETSAT sea ice concentration climate data record," *Cryosphere*, vol. 10, no. 5, pp. 2275–2290, 2016, doi:

10.5194/tc-10-2275-2016.
[18] J. Karvonen, "Evaluation of the operational SAR based Baltic Sea ice concentration products," *Adv. Sp. Res.*, vol. 56, no. 1, pp. 119–132, 2015, doi: 10.1016/j.asr.2015.03.039.
[19] L. Istomina *et al.*, "Melt pond fraction and spectral sea ice albedo retrieval from MERIS data - Part 1: Validation against in situ, aerial, and ship cruise data," *Cryosphere*, vol. 9, no. 4, pp. 1551–1566, 2015, doi: 10.5194/tc-9-1551-2015.
[20] T. Alekseeva *et al.*, "Comparison of Arctic Sea Ice concentrations from the NASA team, ASI, and VASIA2 algorithms with summer and winter ship data," *Remote Sens.*, vol. 11, no. 21, 2019, doi: 10.3390/rs11212481.
[21] L. Kaleschke *et al.*, "SMOS sea ice product: Operational application and validation in the Barents Sea marginal ice zone," *Remote Sens. Environ.*, vol. 180, pp. 264–273, 2016, doi: 10.1016/j.rse.2016.03.009.
[22] G. Aulicino, P. Wadhams, and F. Parmiggiani, "SAR Pancake Ice thickness retrieval in the Terra Nova Bay (Antarctica) during the PIPERS expedition in Winter 2017," *Remote Sens.*, vol. 11, no. 21, 2019, doi: 10.3390/rs11212510.
[23] J. C. Comiso, S. F. Ackley, and A. L. Gordon, "Antarctic sea ice microwave signatures and their correlation with in situ ice observations (Weddell Sea).," *J. Geophys. Res.*, vol. 89, no. C1, pp. 662–672, 1984, doi: 10.1029/JC089iC01p00662.
[24] M. A. Knuth and S. F. Ackley, "Summer and early-fall sea-ice concentration in the Ross Sea: Comparison of in situ ASPeCt observations and satellite passive microwave estimates," in *Annals of Glaciology*, 2006, vol. 44, pp. 303–309, doi: 10.3189/172756406781811466.
[25] G. Spreen, L. Kaleschke, and G. Heygster, "Sea ice remote sensing using AMSR-E 89-GHz channels," *J. Geophys. Res. Ocean.*, vol. 113, no. 2, 2008, doi: 10.1029/2005JC003384.
[26] A. Beitsch, S. Kern, and L. Kaleschke, "Comparison of SSM/I and AMSR-E sea ice concentrations with ASPeCt ship observations around antarctica," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 1985–1996, 2015, doi: 10.1109/TGRS.2014.2351497.
[27] B. Ozsoy-Cicek, S. F. Ackley, A. Worby, H. Xie, and J. Lieser, "Antarctic sea-ice extents and concentrations: Comparison of satellite and ship measurements from International Polar Year cruises," *Ann. Glaciol.*, vol. 52, no. 57 PART 2, pp. 318–326, 2011, doi: 10.3189/172756411795931877.
[28] "IceWatch/ASSIST (Arctic Ship-Based Sea Ice Standardization)," *Norwegian Meteorological Institute*. [Online]. Available: https://icewatch.met.no/.
[29] "ASPeCt (Antarctic Sea Ice Processes and Climate)," *Scientific Committee on Antarctic Research (SCAR)*. [Online]. Available: http://aspect.antarctica.gov.au/.
[30] A. Singh, H. Kalke, M. Loewen, and N. Ray, "River Ice Segmentation With Deep Learning," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–10, 2020, doi: 10.1109/tgrs.2020.2981082.
[31] X. Zhang *et al.*, "ICENET: A semantic segmentation deep network for river ice by fusing positional and channel-wise attentive features," *Remote Sens.*, vol. 12, no. 2, pp. 1–22, 2020, doi: 10.3390/rs12020221.
[32] S. Ansari, C. D. Rennie, O. Seidou, J. Malenchak, and S. G. Zare, "Automated monitoring of river ice processes using shore-based imagery," *Cold Reg. Sci. Technol.*, vol. 142, pp. 1–16, Oct. 2017, doi: 10.1016/j.coldregions.2017.06.011.
[33] R. Prabha, M. Tom, M. Rothermel, E. Baltsavias, L. Leal-Taixe, and K. Schindler, "Lake Ice Monitoring with Webcams and Crowd-Sourced Images," Feb. 2020.
[34] B. Weissling, S. Ackley, P. Wagner, and H. Xie, "EISCAM - Digital image acquisition and processing for sea ice parameters from ships," *Cold Reg. Sci. Technol.*, vol. 57, no. 1, pp. 49–60, 2009, doi: 10.1016/j.coldregions.2009.01.001.
[35] Q. Zhang and R. Skjetne, "Image Techniques for Identifying Sea-Ice Parameters," *Identif. Control*, vol. 35, no. 4, pp. 293–301, 2014, doi: 10.4173/mic.2014.4.6.
[36] Q. Zhang, R. Skjetne, and B. Su, "Automatic image segmentation for boundary detection of apparently connected sea-ice floes," in *The proceedings of the 22nd International Conference on Port and Ocean Engineering under Arctic Conditions*, 2013.
[37] E. Kim, G. S. Dahiya, S. Løset, and R. Skjetne, "Can a computer see what an ice expert sees? Multilabel ice objects classification with convolutional neural networks," *Results Eng.*, vol. 4, 2019, doi: 10.1016/j.rineng.2019.100036.
[38] O.-M. Pedersen and E. Kim, "Arctic Vision: Using Neural Networks for Ice Object Classification, and Controlling How They Fail," *J. Mar. Sci. Eng.*, vol. 8, no. 10, p. 770, 2020, doi: 10.3390/jmse8100770.
[39] B. Dowden, O. De Silva, W. Huang, and D. Oldford, "Sea Ice Classification via Deep Neural Network Semantic Segmentation," *IEEE Sens. J.*, pp. 1–1, Oct. 2020, doi: 10.1109/jsen.2020.3031475.
[40] N. Panchi, E. Kim, and A. Bhattacharyya, "Understanding polar environment: Preliminary results from deep-learning-based segmentation of optical ice images," in *RINA, Royal Institution of Naval Architects - 6th International Conference on Ship and Offshore Technology, Papers*, 2019, pp. 169–179.
[41] Y. Zhu *et al.*, "Improving semantic segmentation via video propagation and label relaxation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019, vol. 2019-June, pp. 8848–8857, doi: 10.1109/CVPR.2019.00906.
[42] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Communications of the ACM*, 2017, vol. 60, no. 6, pp. 84–90, doi: 10.1145/3065386.
[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[44] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.
[45] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 11211 LNCS, pp. 833–851, doi: 10.1007/978-3-030-01234-2_49.
[46] 5 th Session of JCOMM Expert Team on Sea Ice, "Sea Ice Nomenclature," 2014.
[47] S. Kim, "label-tool." www.github.com, 2019.
[48] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and Recognition Using Structure from Motion Point Clouds," in *ECCV (1)*, 2008, pp. 44–57.
[49] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene Parsing through ADE20K Dataset," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[50] M. Cordts *et al.*, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
[51] P. Szymański and T. Kajdanowicz, "Scikit-multilearn: A scikit-based Python environment for performing multi-label classification," *J. Mach. Learn. Res.*, vol. 20, Feb. 2019, doi: 10.5281/zenodo.3670933.
[52] K. Sechidis, G. Tsoumakas, and I. Vlahavas, "On the stratification of multi-label data," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6913 LNAI, no. PART 3, pp. 145–158, 2011, doi: 10.1007/978-3-642-23808-6_10.
[53] P. Szymański and T. Kajdanowicz, "A Network Perspective on Stratification of Multi-Label Data," in *Proceedings of the First International Workshop on Learning with Imbalanced Domains: Theory and Applications*, 2017, vol. 74, pp. 22–35.
[54] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, vol. 2017-Janua, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.
[55] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified Perceptual Parsing for Scene Understanding," *CoRR*, vol. abs/1807.1, 2018.
[56] P. Wang *et al.*, "Understanding Convolution for Semantic Segmentation," *Proc. - 2018 IEEE Winter Conf. Appl. Comput. Vision, WACV 2018*, vol. 2018-Janua, pp. 1451–1460, 2018, doi: 10.1109/WACV.2018.00163.
[57] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
[58] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters - Improve semantic segmentation by global convolutional network,"

Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol. 2017-Janua, pp. 1743–1751, 2017, doi: 10.1109/CVPR.2017.189.

[59] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation," *CoRR*, vol. abs/1606.0, 2016.

[60] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017, doi: 10.1109/TPAMI.2016.2644615.

[61] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[62] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, doi: 10.1109/CVPR.2017.106.

[63] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely Connected Convolutional Networks," *CoRR*, vol. abs/1608.0, 2016.

[64] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv Prepr. arXiv1412.6980*, 2014.

[66] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1--learning rate, batch size, momentum, and weight decay," *arXiv Prepr. arXiv1803.09820*, 2018.

[67] J. Howard and S. Gugger, "Fastai: A Layered API for Deep Learning," *Information*, vol. 11, no. 2, p. 108, Feb. 2020, doi: 10.3390/info11020108.

[68] A. Paszke *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[69] Yassine, "Semantic Segmentation in PyTorch." www.github.com, 2019.

[70] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with Gaussian edge potentials," in *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, 2011.

[71] Junnan Zhang and Hanyi Nie, "A post-processing method based on Fully connected CRFs for chronic wound images segmentation and identification," 2018, pp. 21–29, doi: 10.5121/csit.2018.81703.

[72] K. Kamnitsas *et al.*, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, 2017.

[73] M. T. T. Teichmann and R. Cipolla, "Convolutional CRFs for Semantic Segmentation," *CoRR*, vol. abs/1805.0, May 2018.

[74] Z. Zhang, "ExFuse: Enhancing Feature Fusion for Semantic Segmentation."

[75] L.-C. Chen *et al.*, "Searching for Efficient Multi-Scale Architectures for Dense Image Prediction."

[76] H. Zhang, H. Zhang, C. Wang, and J. Xie, "Co-occurrent Features in Semantic Segmentation."

[77] X. Zhang, H. Xu, H. Mo, J. Tan, C. Yang, and W. Ren, "DCNAS: Densely Connected Neural Architecture Search for Semantic Image Segmentation," Mar. 2020.

[78] M. Cordts *et al.*, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, doi: 10.1109/CVPR.2016.350.

[79] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognit. Lett.*, 2009, doi: 10.1016/j.patrec.2008.04.005.

[80] X. Nie, M. Duan, H. Ding, B. Hu, and E. K. Wong, "Attention Mask R-CNN for ship detection and segmentation from remote sensing images," *IEEE Access*, 2020, doi: 10.1109/ACCESS.2020.2964540.

[81] O. Oktay *et al.*, "Attention U-Net: Learning where to look for the pancreas," *arXiv*. 2018.

[82] A. Tao, K. Sapra, and B. Catanzaro, "Hierarchical multi-scale attention for semantic segmentation," *arXiv*. 2020.

[83] O.-M. Pedersen and E. Kim, "Evaluating Human and Machine Performance on the Classification of Sea Ice Images," in *Proceedings of the 25th International Symposium on Ice*, 2020.

[84] Michelle E Johnston; G W Timco; Canadian Hydraulics Centre., *Understanding and identifying old ice in summer*. [Ottawa] : National Research Council Canada, Canadian Hydraulics Centre, 2008.

[85] A. Cheng, B. Casati, A. Tivy, T. Zagon, J. F. Lemieux, and L. Bruno Tremblay, "Accuracy and inter-analyst agreement of visually estimated sea ice concentrations in Canadian Ice Service ice charts using single-polarization RADARSAT-2," *Cryosphere*, 2020, doi: 10.5194/tc-14-1289-2020.

TABLE I
CLASSES AND THEIR DESCRIPTIONS

No.	Class	Description
1	Broken ice	Predominantly flat ice cover broken by gravity waves or due to melting decay.
2	Brash ice	Accumulations of floating ice made up of fragments not more than 2 m across, the wreckage of other forms of ice.
3	Deformed ice	A general term for ice that has been squeezed together and, in places, forced upwards and downwards). Subdivisions are rafted ice, ridged ice, and hummocked ice.
4	Floeberg	A large piece of sea ice composed of a hummock, or a group of hummocks frozen together and separated from any ice surroundings. It typically protrudes up to 5 m above sea level.
5	Floebit	A relatively small piece of sea ice, typically not more than 10 m across, composed of a hummock (or more than one hummock) or part of a ridge (or more than one ridge) frozen together and separated from any surroundings. It typically protrudes up to 2 m above sea level.
5	Iceberg	A piece of glacier origin floating at sea.
6	Ice floe	Any contiguous piece of sea ice.
7	Level ice	Sea ice that has not been affected by deformation.
8	Pancake ice	Predominantly circular pieces of ice from 30 cm - 3 m in diameter, and up to approximately 10 cm in thickness, with raised rims due to the pieces striking against one another.
9	Open water	A large area of freely navigable water in which sea-ice is present in concentrations less than 1/10.
10	Melt ponds	Pools of water that formed on top of the sea-ice in the warmer months of spring and summer.
11	Underwater ice	The submerged but visible part of any sea-ice object.
12	Broken ice	Predominantly flat ice cover broken by gravity waves or due to melting decay.

Classes from 1-10 are taken or adapted from the WMO's sea-ice nomenclature [46]. Classes Sky and Shore are self-explanatory.

TABLE II
STATISTICS FOR CHANNEL WISE NORMALIZATION OF IMAGES

Channel	Mean	Standard deviation
Red (R)	0.485	0.229
Green (G)	0.456	0.224
Blue (B)	0.406	0.225

Each of the RGB values was normalized using the mean and standard deviation of the corresponding channel. These values were derived from the ImageNet dataset [64].

TABLE III
HYPERPARAMETERS FOR TRAINING NEURAL NETWORKS

Parameter	Stage 1	Stage 2

α_{max}	5 x 10^{-4} for the unfrozen part of the network (decoder)	0.125 x 10^{-4} for the decoder, 1.25 x 10-4 for the encoder
% of iterations till which α increases	90%	90%
Epochs	20	60
β_1	0.9	0.9
β_2	0.99	0.99
Weight decay	10-2	10-2

α is the learning rate, and β_1, β_2, and weight decay are parameters of the Adam optimizer [65].

TABLE IV
ALGORITHM FOR THE 'MEAN' ENSEMBLE

Input: *3 channeled RGB image (img) [512 × 512 × 3]*

- *Pass the image through the neural networks:*
  Output1 [512 × 512 × 14] = PSPNet(img)
  Output2 [512 × 512 × 14] = PSPNet(img)
  Output3 [512 × 512 × 14] = DeepLabV3+(img)
  Output4 [512 × 512 × 14] = UPerNet(img)
- *Combine the outputs:*
  *Combined output [512 × 512 × 14] = Mean(Output1, Output2, Output3, Output4)**
- *Postprocessing:*
  Postprocessed output [512 × 512 × 14] = ConvCRF(img, Combined output)
- *Take the index of the max. value along the 3rd dim:*
  Segmentation mask [512 × 512 × 1] = Argmax(Postprocessed output)

Output: *1 channeled segmentation mask [512 × 512 × 3]*

*There can be several other ways to combine the neural network outputs

TABLE V
DEFINITIONS FOR THE PERFORMANCE METRICS

		Predicted	
		True	False
Ground Truth	True	*TP*	*FN*
	False	*FP*	*TN*

True Positives (*TP*), False Positives (*FP*), True Negatives (*TN*), and False Negatives (*FN*), determined over the whole data set (i.e., validation set or test set). Here, one pixel is considered as a sample. In the case of a particular class (*c*), the TP_c, TN_c, FP_c, and FN_c are calculated for that particular class.

TABLE VI
AVERAGE PER-CLASS CONFUSION FOR THE PROPOSED APPROACH

		True class		
		Floeberg	Floebit	Melt pond
Test	Grayscale	84% predicted as Deformed ice	37% predicted as Ice floe	40% predicted as Ice floe
	Vignette	66% predicted as Deformed ice	89% predicted as Sky	37% predicted as Level ice
	Clear	85% predicted as Deformed ice	30% predicted as Iceberg	51% predicted as Ice floe
Validation	Val-1	64% predicted as Deformed ice	39% predicted as Ice floe	64% predicted as Underwater ice
	Val-2	21% predicted as Deformed ice	73% predicted as Ice floe	84% predicted as Ice floe
	Val-3	93% predicted as Iceberg	71% predicted as Ice floe	76% predicted as Ice floe
	Val-4	35% predicted as Deformed ice	21% predicted as Iceberg	82% predicted as Underwater ice
	Val-5	7% predicted as Deformed ice	27% predicted as Brash ice	27% predicted as Deformed ice

Values are for the proposed segmentation approach (Mean ensemble + ConvCRF postprocessing) for each of the five validation sets and three variations of the test set. Each cell presents how much percentage of the true class was falsely predicted as something else.

TABLE VII
MEAN IOU VALUES FOR ICE OBJECTS AND NONICE OBJECTS

Dataset	Ice objects	Nonice objects
Avg. of 5-fold cross validation	0.642	0.573
Clear Test	0.502	0.544
Grayscale Test	0.422	0.512
Vignette Test	0.339	0.297

Values are for the proposed approach (Mean Ensemble + ConvCRF postprocessing)

TABLE VIII
PERFORMANCE METRICS FOR ICE OBJECT VS. NONICE OBJECT SEGMENTATION (BINARY SEGMENTATION PROBLEM)

Dataset	Mean IOU	Accuracy	F1 score
Avg. of 5-fold cross validation	0.933	0.957	0.965
Clear Test	0.948	0.963	0.973
Grayscale Test	0.946	0.961	0.972
Vignette Test	0.758	0.829	0.862

Values are for the proposed approach (Mean Ensemble + ConvCRF postprocessing)

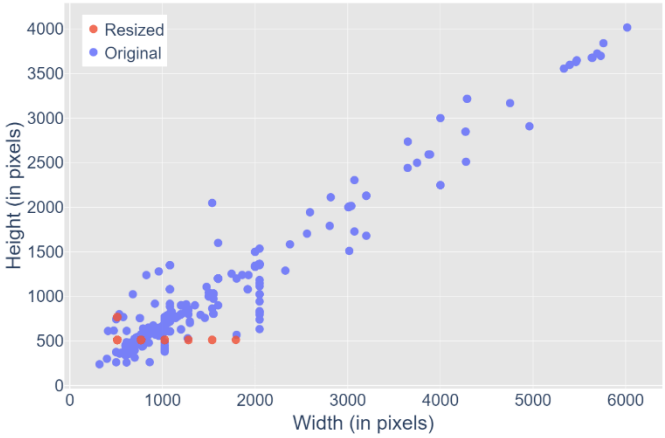


Fig. 1. Resolutions of images before and after resizing.

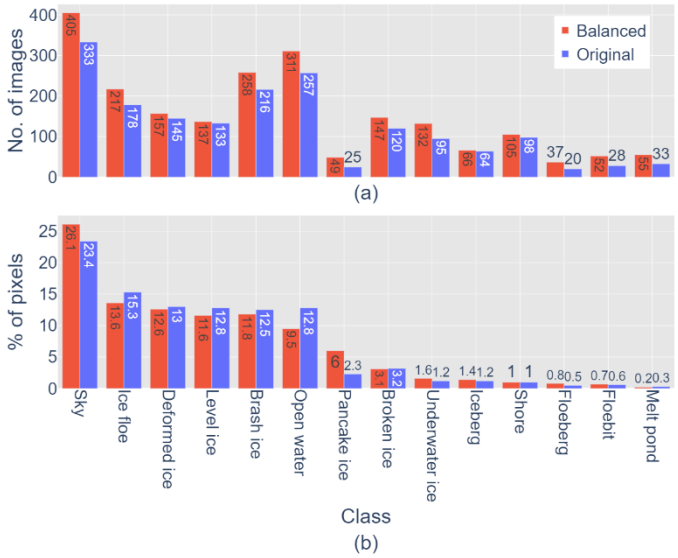


Fig. 2. (a) No. of images and (b) The percentage of total pixels per class for both original and balanced dataset. The balanced dataset was obtained through preprocessing, i.e., resizing and oversampling.

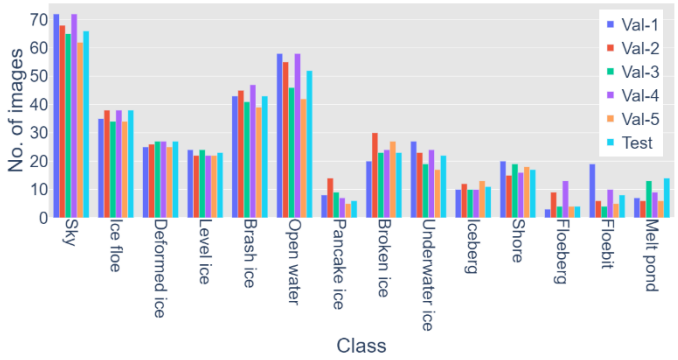
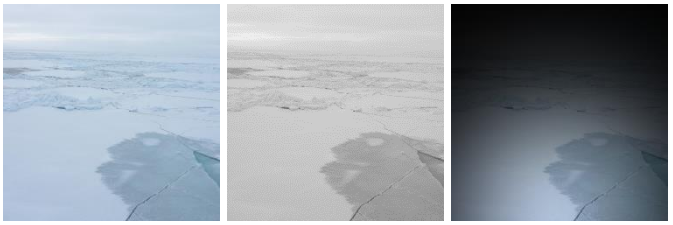


Fig. 3. Per-class distribution of images for the six parts of the dataset (5 validation, i.e., Val-1, …, Val-5 + 1 Test).



(a) Clear (b) Grayscale (c) Vignette effect

Fig. 4. Samples from the three variations of the test set.

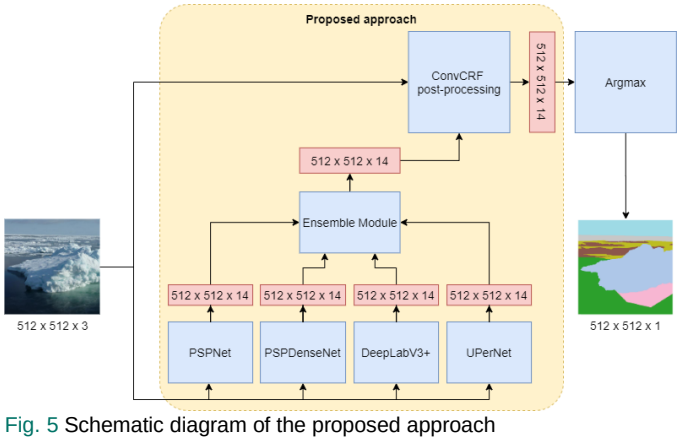
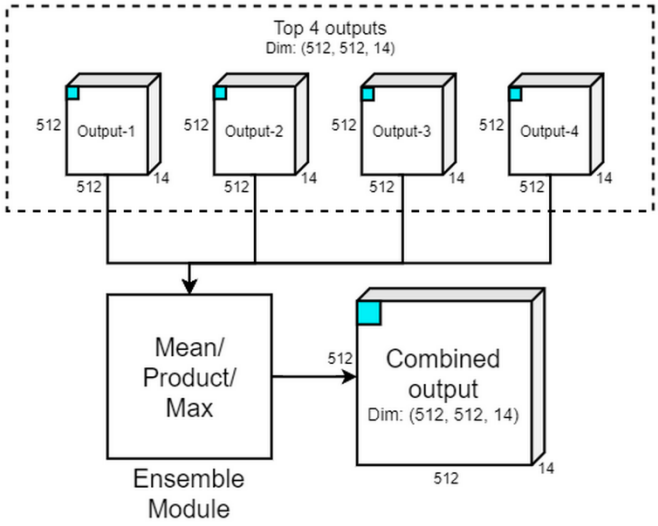
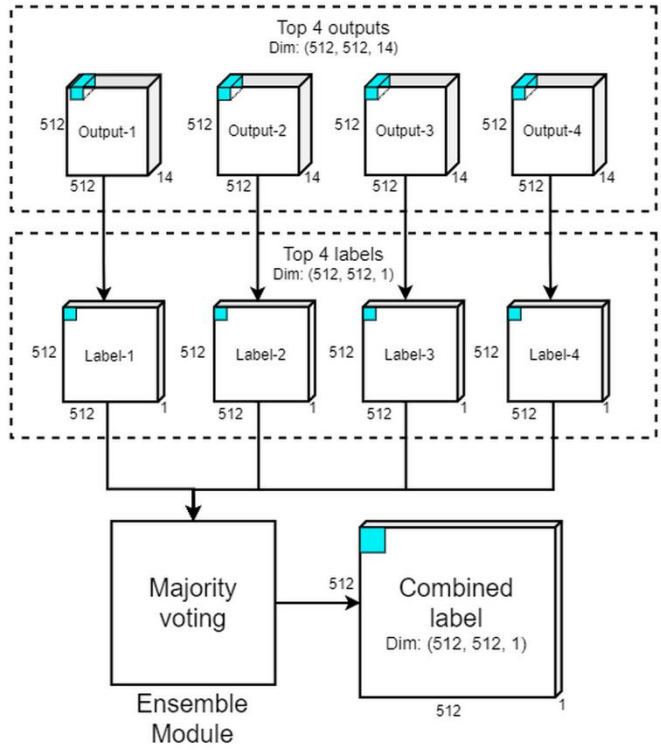


Fig. 5 Schematic diagram of the proposed approach



(a) Category 1: Combination method of Mean/Product/Max to combine each of the values from the Output 1,2,3,4. The blue pixels in the Outputs are combined to give the value of the blue pixel in the Combined output. This holds true for all such pixels in the Combined output tensor. The top 4 outputs and the combined output have a dimension of 512x512x14, i.e., a grid size of 512x512 (equal to the dimension of the input image) and 14 channels for 14 classes.



(b) Category 2: Combination method of Majority voting to vote between the values from label 1,2,3,4. The argmax of the blue pixels in the output-i gives the blue pixel in the label-i, and the blue pixels in label 1,2,3,4 combine to give the blue pixel in the combined label. This holds true for all such pixels in the Combined label tensor. The top 4 outputs have a dimension of 512x512x14, while the top 4 labels and combined label have a dimension of 512x512x1. Here, 512x512 is the grid size (equal to the dimension of input image), 14 is the number of channels for 14 classes and 1 is the number of channels for labels since labels are a single value from 1 to 14.

Fig. 6. Ensemble Modules.

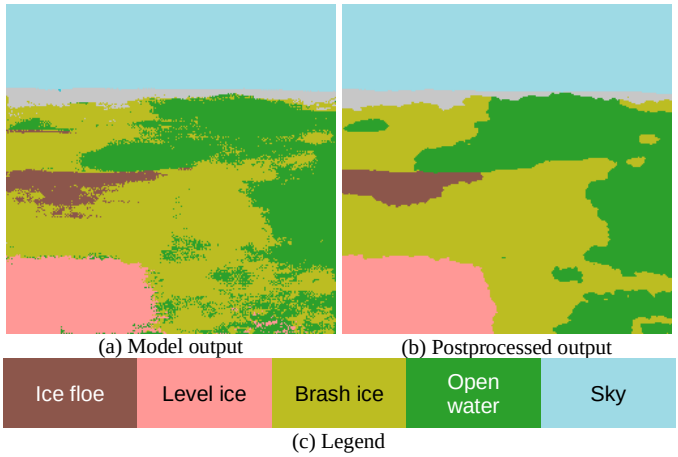


Fig. 7. Effect of postprocessing on the model output

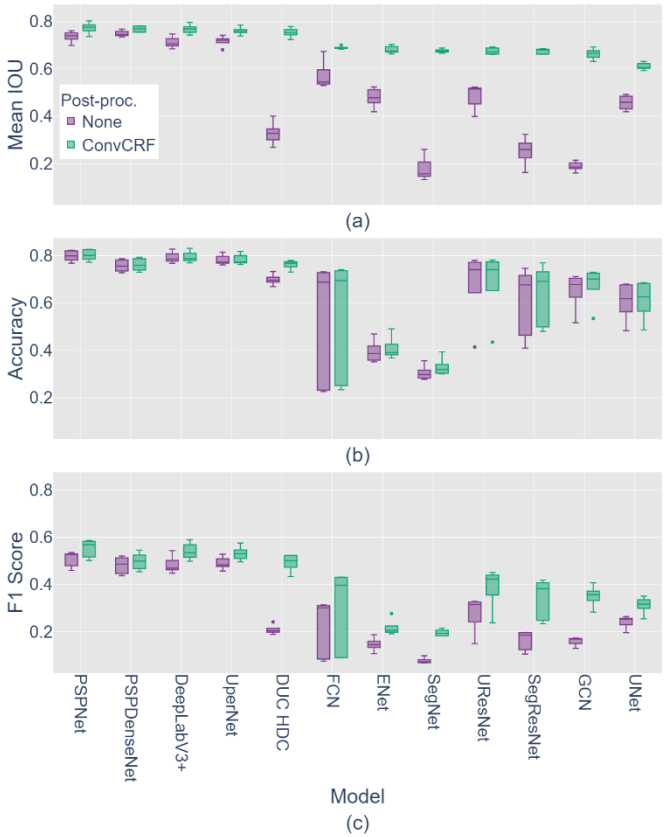


Fig. 8. Performance metrics for the 5-fold cross-validation of single model pipelines. For every model on the X-axis, we plot two boxes for each of the three metrics (three plots). The first box (purple) presents the spread (over five validation folds) of the metric for the model outputs without any postprocessing, and the second box (green) presents the spread of the metric for the postprocessed outputs (using ConvCRF postprocessing). The X-axis is sorted based on the averaged (over five validation folds) mean IOU value of the postprocessed results.

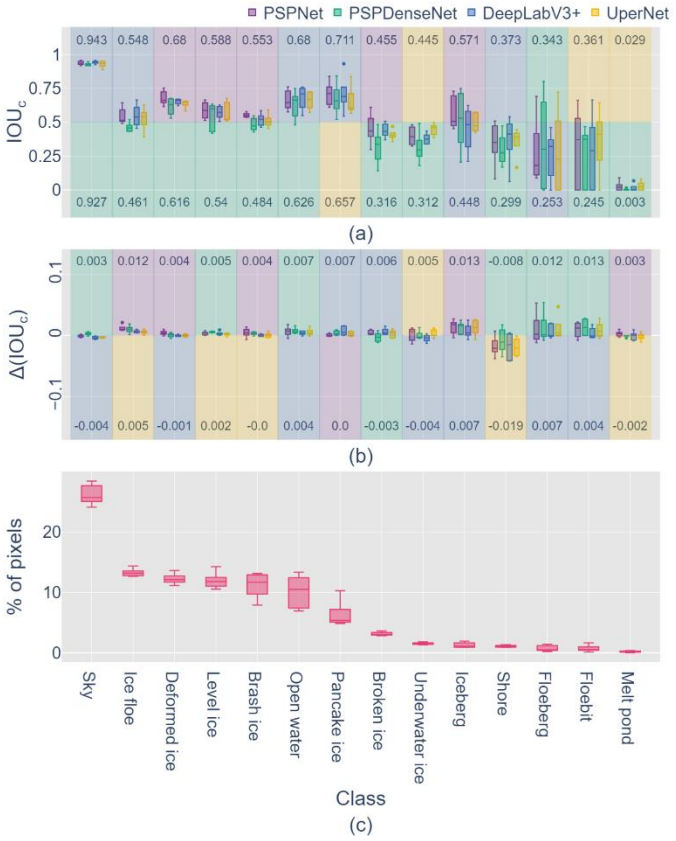


Fig. 9. (a) Per-class IOU (after postprocessing) for the top four single model pipelines (b) Change in per-class IOU due to ConvCRF postprocessing. $\Delta(IOU_c) = IOU_c$ of the postprocessed results – IOU_c of the non-postprocessed results. (c) The percentage of total pixels for the validation sets (1 pixel = 1 sample, from the point of view of calculation of metrics). The boxes represent the range of values over five validation folds. The background colors in the top 2 plots represent the model with average (over five validation folds) highest and average lowest values in the top half and bottom half, respectively. The average highest and average lowest values are also mentioned at the top and bottom, respectively. The X-axis is sorted based on the average values (over five validation folds) of the percentage of total pixels per class.

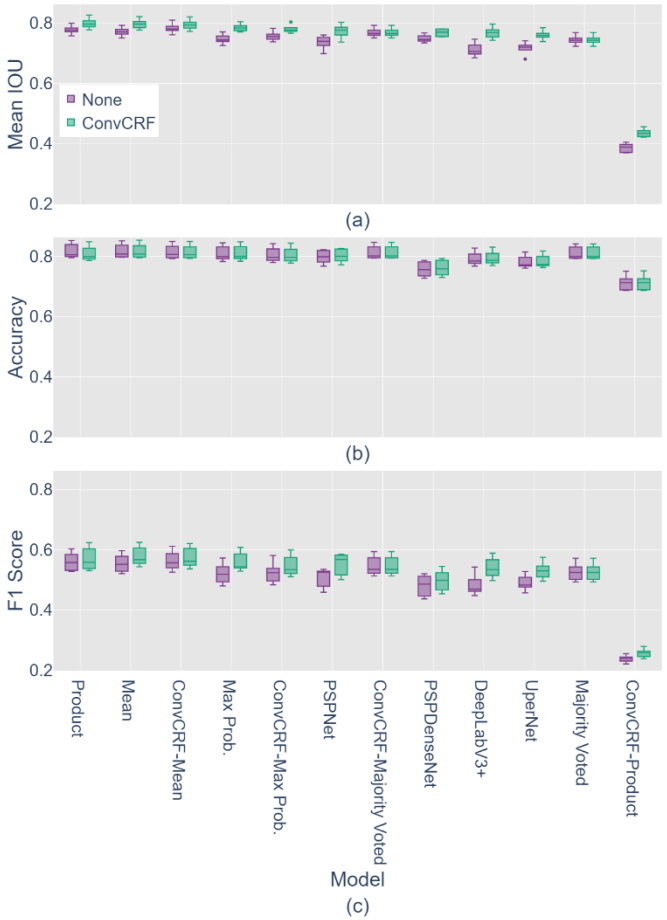


Fig. 10. Performance metrics from the 5-fold cross-validation for eight different ensemble pipelines and the top four single-model pipelines. For every model on the X-axis, we plot two boxes for each of the three metrics. The first box (purple) presents the spread (over five validation folds) of the metric for the model outputs without any postprocessing, and the second box (green) presents the spread of the metric for the postprocessed outputs (using ConvCRF postprocessing). The X-axis is sorted based on the averaged (over five validation folds) mean IOU value of the postprocessed results.

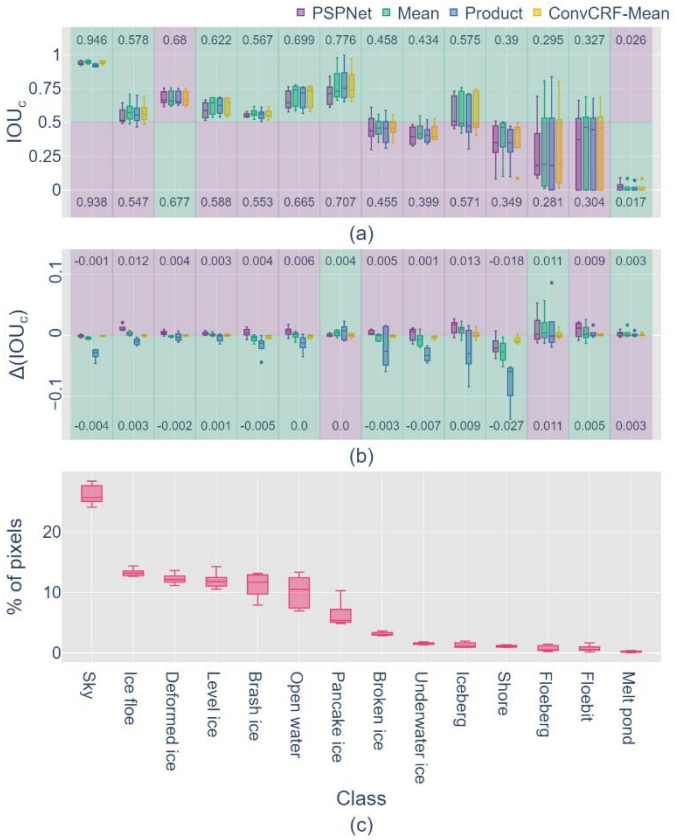


Fig. 11. (a) Per-class IOU (after postprocessing) for the top three ensembles (Product, Mean, ConvCRF-Mean) and the best model (PSPNet). (b) Change in per-class IOU due to ConvCRF postprocessing. $\Delta(IOU_c) = IOU_c$ of the postprocessed results – IOU_c of the non-postprocessed results. (c) The percentage of total pixels for the validation sets (1 pixel = 1 sample, from the point of view of calculation of metrics). The boxes represent the range of values over five validation folds. The background colors in the top half of the top 2 plots represent which among the mean ensemble (green) and single-model pipeline (purple) achieved a higher value. Similarly, the background colors in the bottom half of the top 2 plots represent which one of the above two pipelines achieved a lower value. The values for these two segmentation pipelines have also been mentioned on the respective background color. The X-axis is sorted based on the average values (over five validation folds) of the total pixels per class.

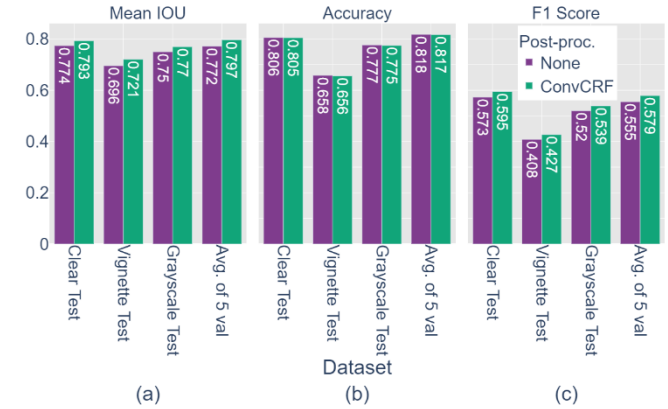


Fig. 12. Performance metrics from the three variations of the test set (test set with clear images (clear test), test set with vignette effect (vignette test), and test set with grayscale images (grayscale test)) and 5-fold cross-validation for the proposed ensemble approach (mean). For every dataset on the X-axis, we plot two bars for each of the three metrics. The first bar (purple) presents the metric score for the ensemble outputs without any postprocessing, and the second bar (green)

presents the metric score for the postprocessed outputs (using ConvCRF postprocessing).

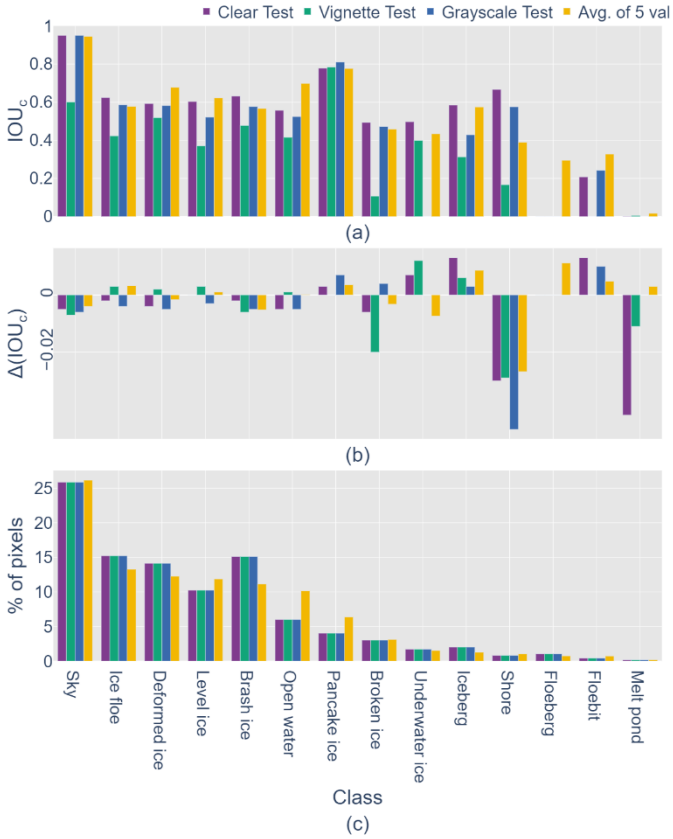


Fig. 13. Per-class performance on the three test variations of the test set (Test set with clear images (Clear Test), test set with vignette effect (Vignette Test), and test set with grayscale images (Grayscale Test)) and 5-fold cross-validation of the proposed ensemble approach (Mean). (a) Per-class IOU (after postprocessing) from the test and validation sets for the proposed ensemble approach (Mean ensemble + postprocessing). (b) Change in per-class IOU due to ConvCRF postprocessing. $\Delta(IOU_c) = IOU_c$ of the postprocessed results – IOU_c of the nonprocessed results. (c) The percentage of total pixels per class for the test and validation sets.

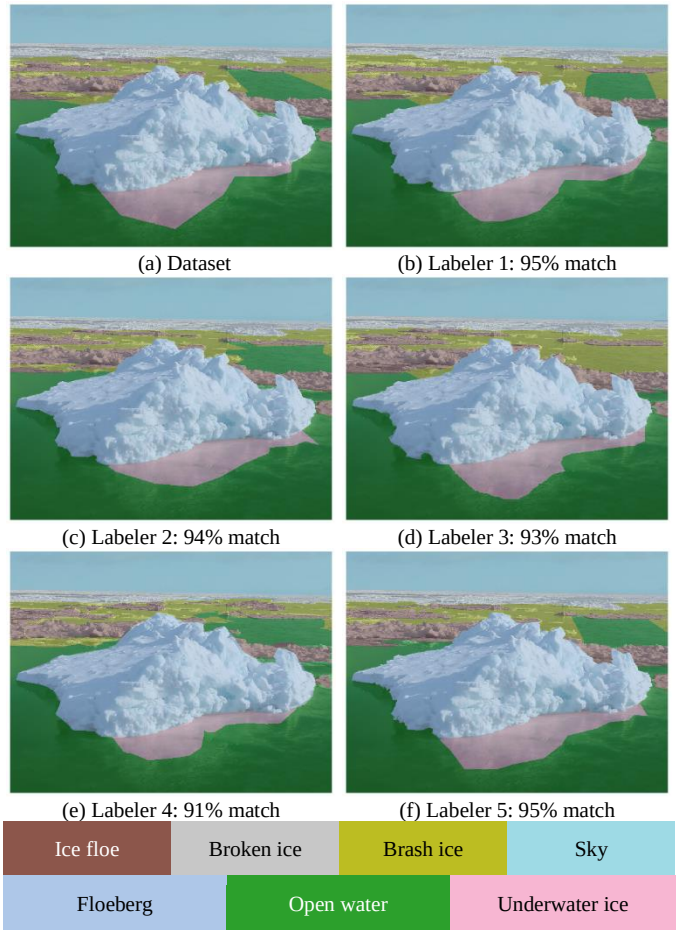
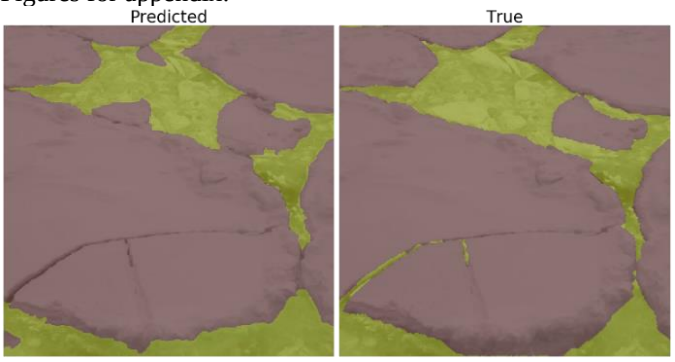
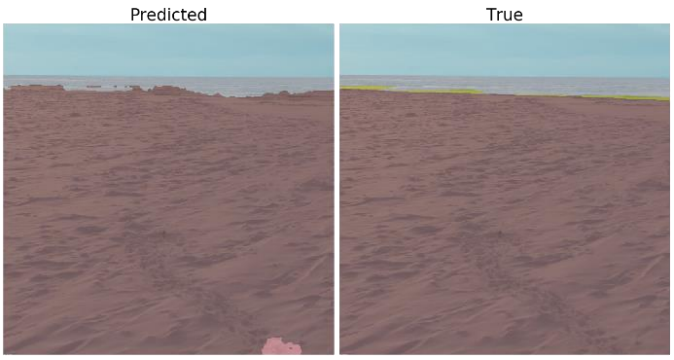


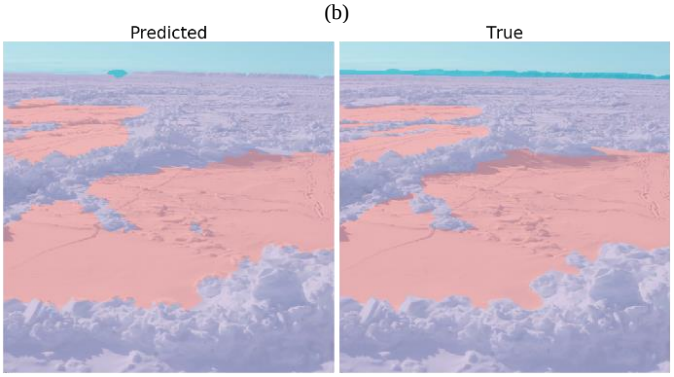
Fig. 14. Labels used for comparison. The top-left label is from the dataset, while the rest were labeled by different labellers and were only for comparison. The % match is with respect to the label from the dataset.

Figures for appendix:

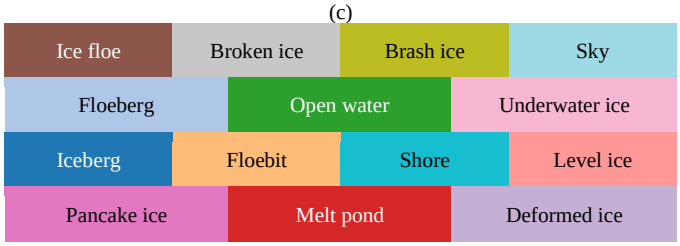




(b)



(c)



(e) Legend

Fig. 15. Qualitative results from the proposed ensemble approach (i.e., Mean ensemble + ConvCRF postprocessing).

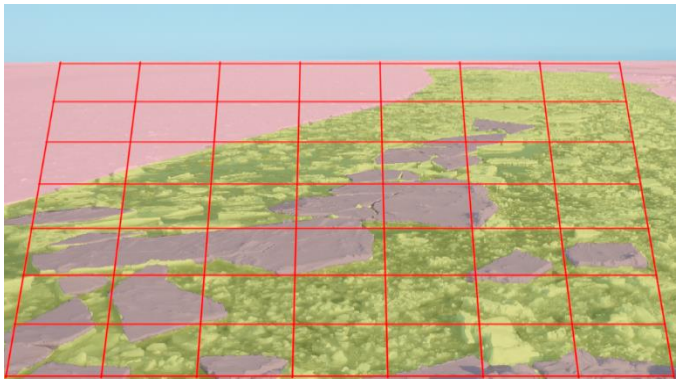


Fig. 16. Sample labeled image with a mesh for ice concentration and floe size measurement (Demo only).