remote sensing

MDPI

Article

The Identification of Ice Floes and Calculation of Sea Ice Concentration Based on a Deep Learning Method

Li Zhou [1], Jinyan Cai [2] and Shifeng Ding [2,*]

[1] School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2] School of Naval Architecture and Ocean Engineering, Jiangsu University of Science and Technology, Zhenjiang 212100, China
* Correspondence: 15001945469@163.com

Abstract: When navigating ships in cold regions, sea ice concentration plays a crucial role in determining a ship's navigability. However, automatically extracting the sea ice concentration and floe size distribution remains challenging, due to the difficulty in detecting all the ice floes from the images captured in complex polar environments, particularly those that include both ships and sea ice. In this paper, we propose using the YOLACT network to address this issue. Cameras installed on the ship collect images during transit and an image dataset is constructed to train a model that can intelligently identify all the targets in the image and remove any noisy targets. To overcome the challenge of identifying seemingly connected ice floes, the non-maximum suppression (NMS) in YOLACT is improved. Binarization is then applied to process the detection results, with the aim of obtaining an accurate sea ice concentration. We present a color map and histogram of the associated floe size distribution based on the ice size. The speed of calculating the sea ice density of each image reaches 21 FPS and the results show that sea ice concentration and floe size distribution can be accurately measured. We provide a case study to demonstrate the effectiveness of the proposed approach.

Keywords: deep learning; ice floe identification; image processing; floe size distribution; ice concentration; YOLACT

check for updates

Citation: Zhou, L.; Cai, J.; Ding, S. The Identification of Ice Floes and Calculation of Sea Ice Concentration Based on a Deep Learning Method. *Remote Sens.* **2023**, *15*, 2663. https://doi.org/10.3390/rs15102663

Academic Editors: Yi Luo and Mohammed Shokr

Received: 21 April 2023
Revised: 17 May 2023
Accepted: 18 May 2023
Published: 19 May 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the increasing exploration and exploitation of hydrocarbons in ice-covered waters, more attention is being paid to marine operations impacted by ice [1]. The greatest challenge to Arctic navigation is sea ice, which covers approximately 7% of the world's oceans [2]. Sea ice concentration and floe size distribution are the key parameters that significantly impact navigability. In cold regions, sea ice comes in various sizes, ranging from meters to kilometers in size, and an icebreaker traveling through an area of high sea ice concentration or large floe size must consider whether the ice conditions can support safe passage. Therefore, real-time monitoring and an identification of the size and shape of floes contribute to automatically detecting hazardous conditions [3,4] and the auxiliary decision making for ship navigation in cold regions. Additionally, this floe size distribution is crucial in forming a digital ice field. In numerical simulations of ice loads in a broken ice field [5–11], the floe size distribution in the digital ice field is approximately estimated based on the ice tank test. If computer vision is used to measure the actual floe size distribution of the broken ice field, the accuracy of the numerical simulations will improve.

However, the retrieval of sea ice information from observational data sets is still limited due to noise from the complex polar environment and a lack of efficient equipment and algorithms for processing cold region images. Therefore, the development of technology for high-resolution field observations and estimations of ice floe size distribution is necessary and meaningful.

Using optical images taken by ship-based cameras is one of the best ways to observe the sea ice conditions in cold regions. Digital image processing techniques can reduce uncertainties in the environment and suppress errors regarding objects, obtaining more reliable information on the sea ice conditions [12]. In ice-covered regions, cameras are commonly used as sensors to characterize these ice conditions [13,14]. Ship-based observations and remote sensing are currently the main methods used to obtain images of ice-covered regions. Lu and Li [15] attached cameras to a ship to take photos and obtain the ice concentration and floe size. Zhang and Skjetne [16] installed cameras on an unmanned aerial vehicle (UAV) to collect data on sea ice and measure the ice statistics and properties.

Remote sensing is a crucial tool used in polar sea ice research [17–19]. Synthetic-aperture radar (SAR) imaging plays an essential role in sea ice data analysis, as it is not restricted by environmental factors, enabling the data to be continuously collected in all weather conditions and through the polar night [20]. However, using satellite high-resolution optical instruments incurs high application costs and cannot distinguish open water from pond ice and persistent cloud cover [21]. To address this, Yan et al. [22] used convolutional neural networks (CNN) to extract the sea ice concentration from Global Navigation Satellite Signal Reflectometry (GNSS-R) data and Zhu et al. [23] used machine learning to classify sea ice with GNSS-R data. Meanwhile, sea ice photography conducted with digital cameras is widely used to determine the actual ice conditions encountered by a ship or aircraft [24,25]. Optical sensors are not suitable for use in the dark and harsh weather; however, shipborne photography is cost-effective, automated, and easy to operate, making it a common method used for scientific expeditions in the polar regions.

When retrieving ice statistics from obtained images, the method of digital image processing determines the quality of the ice information. In an actual ice-covered environment, ice floes usually touch each other closely, making it challenging to identify their borders from optical images. Several researchers have tried to mitigate this issue through a boundary detection of the individual ice floes and an ice floe analysis [16,17,26]. Extracting these ice floes from images is also a significant challenge. Zhang et al. [26] used threshold segmentation to divide the image into "object area" and "background area," which could identify the sea ice. However, the threshold needs to be adjusted when processing images captured in different light conditions [27] and the illumination affects the result. There are many deep learning methods that can be used to identify sea ice, such as YOLO proposed by Redmon et al. [28], which can automatically identify the class and location information of ice. YOLACT, proposed by Bolya et al. [29], is a deep convolutional neural network and can recognize different classes at the pixel level.

To calculate the sea ice concentration of the bow area, images captured by a camera in an outward-looking orientation are used. These images typically contain sea ice, water, and ships, making it necessary to distinguish between these three objects and extract the ice. To accomplish this, sea ice images are collected and used to create a dataset for training the YOLACT network. The trained instance segmentation model can distinguish the sea ice and remove the water and ships from the images. Once the mask of sea ice has been extracted, the number of sea ice pixels can be calculated. The real area of the sea ice is proportional to the number of pixels, which can be calculated according to the ratio between the pixel distance and the actual distance. With the acquisition of the real area, the floe size distribution and ice concentration can be calculated from the resulting data.

2. Description of YOLACT Network

First, we utilize the YOLACT network to calculate the sea ice concentration, drawing on previous knowledge of the network and our observations of the sea ice. Next, we detail the construction of the dataset used for the sea ice identification, as well as the process of the model training and testing. Finally, we explain the method used for calculating the ice concentration.

2.1. Model Description

The YOLACT network, proposed by Bolya et al. [29], is an instance segmentation network. Unlike other methods of sea ice identification, YOLACT is a deep learning method that can identify multiple objects in a complex environment. It can detect images taken in different lighting conditions and identify objects other than sea ice. Figure 1 illustrates the YOLACT network, which, similar to other networks, utilizes the multi-scale feature map and Feature Pyramid Network (FPN) technique proposed by Zhao et al. [30]. The backbone extracts features and transfers feature information to the prediction network through the feature pyramid. The network divides complex detection tasks into two parallel processes to speed up the detection. One branch uses ProtoNet to generate a prototype mask and the other approach uses the prediction header to process the position and show the coefficients, which are filtered by the NMS. In this paper, we adjust the NMS in the network to fit the task of sea ice identification, as the common parameters of NMS can lead to missed detections when detecting images with a high sea ice concentration. The obtained result is linearly combined with the original mask to yield the segmentation of each instance.

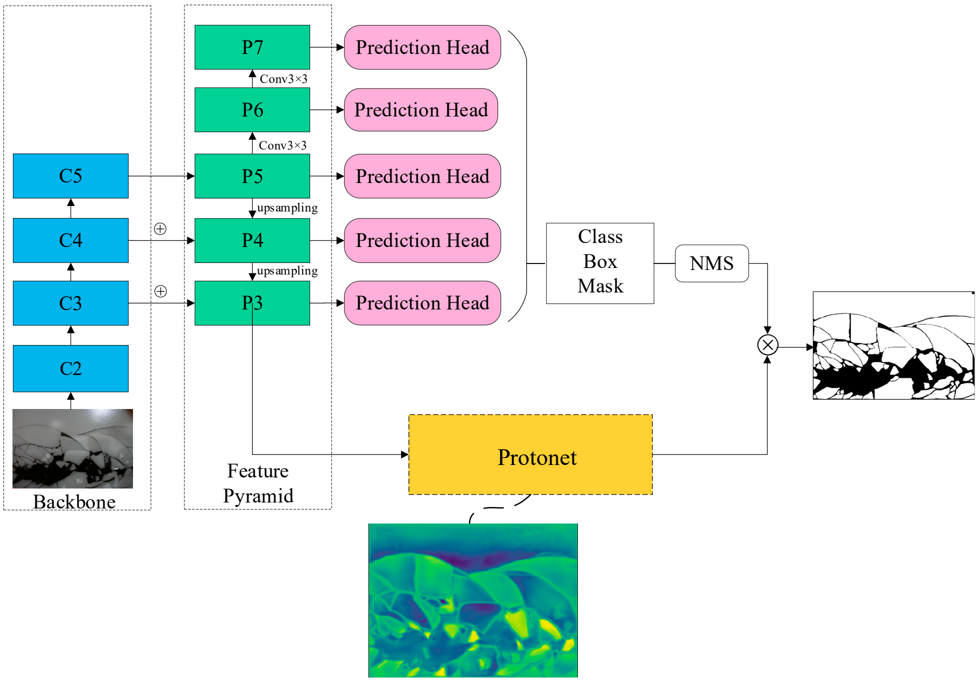


Figure 1. YOLACT framework diagram.

2.2. Dataset for Training Model

To ensure that the trained model can be applied to different environments, it is important not only to select a suitable model, but also to choose images captured in various situations as datasets, such as model ice in an ice tank, sea ice in open water, and images containing both ice breakers and ice in one picture.

The images of the model ice were obtained from the multifunctional ice basin of the Marine Technology Group at Aalto University. The camera captured the model ice images, which usually covered a large area, from the passage above the ice basin. To enhance the applicability of the trained model, we varied the illumination conditions and shooting angles of the camera, as shown in Figure 2. This diversified the dataset to include various detection scenarios, making the trained model suitable for images captured not only from a suitable illumination environment, but also from complex environments and even open water.

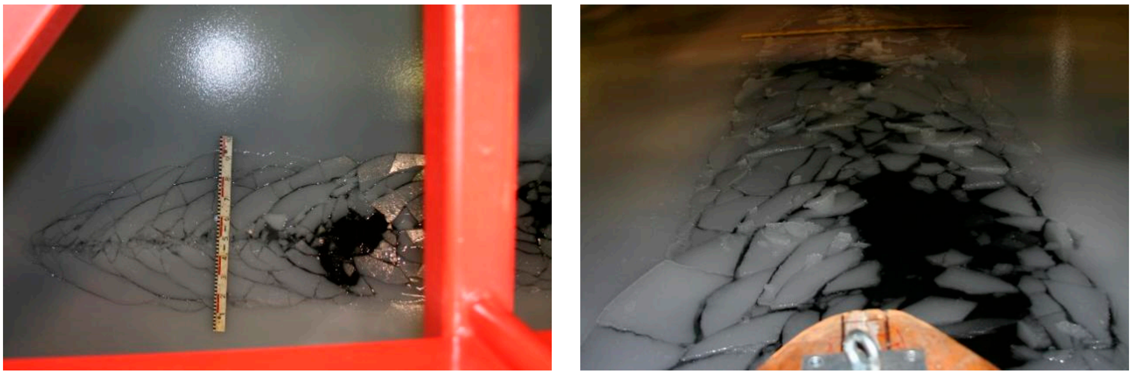

(**a**) (**b**)

Figure 2. Images captured from a complicated environment. (**a**) Image taken against light. (**b**) Image taken with a shooting angle.

During the model testing, we were not required to obtain the ice concentration for the entire ice basin, as this is typically limited to a small area of interest, such as the ice–structure interaction region near the bow. Globally extracting a segmentation of the model ice and background from a whole ice basin image may cause the loss of some ice information, especially when an image suffers from non-uniform illumination or shadow problems. Moreover, sea ice images usually contain multiple ice floes that are closely connected, as shown in Figure 2a, and some parts of floe boundaries can become weaker than others. This problem becomes more challenging with an increasing picture-shooting range. Decreasing the shooting range, however, can result in a loss of information, reducing the amount of valid data for the training set and slowing down the model convergence speed. Moreover, it can enhance noise and affect the segmentation for the detection set. Therefore, we needed to ensure that the images contained sufficient information for creating a good training set and adjust the parameters during the training to minimize the information loss. To make the trained model better suited for sea ice detection, we also included some images of crushing ice fields in the dataset, as illustrated in Figure 3.

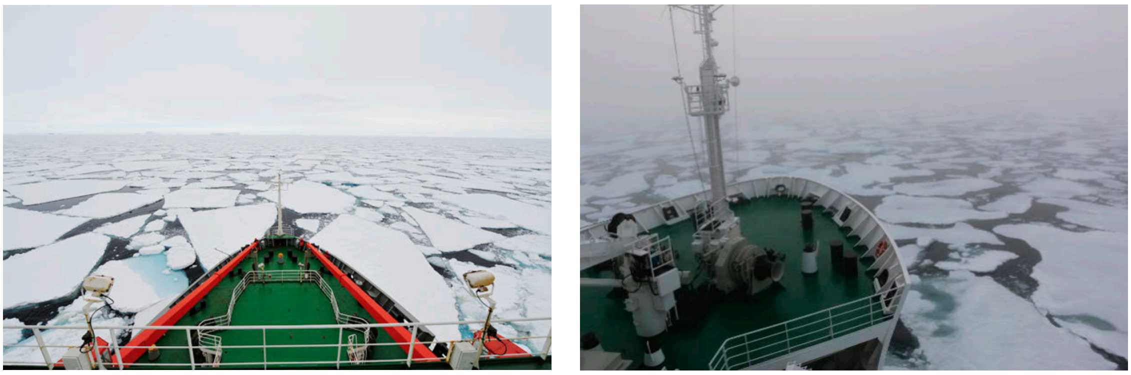

(**a**) (**b**)

Figure 3. Images captured on the ship. (**a**) the image taken on a sunny day. (**b**) the image taken in foggy weather.

To classify the ice, ice breakers, and water in the images, the samples were grouped into two classes: ice and ship, while the water was considered as the background of the image. The purpose of this classification was to determine whether the trained model could differentiate the sea ice and ice breakers from the water, which would enable the mapping of the quantitative sea ice concentration. During all the tests, 80% of the dataset was allocated for training and 20% for validation.

2.3. Model Training and Testing

In this experiment, the YOLACT model used the ResNet50 network and was trained for 40,000 iterations with an initial learning rate of 0.01. The decay was set to occur at the 20,000th, 30,000th, and 35,000th iterations, respectively, with a decay rate of 10% of the current learning rate. Additionally, because the images of the sea ice were taken from high angles, the image size was enhanced to 1000 (normally set to 500). The "image size" parameter refers to the size of the input image into the neural network. The actual size of the images taken from the ice basin could be up to 1200 and the large shooting range captured a significant amount of sea ice. If we continued to set the image size to 550, the small size of ice blocks in the images may not have been clear enough to train the model. For the instance segmentation of sea ice, the main evaluation metric used was average precision (*AP*). To determine whether an object was accurately segmented, the Intersection over Union (IoU) threshold could be used to express the similarity coefficient between the two sets of data. In instance segmentation, IoU represents the overlap area between the predicted sea ice mask M_p and the real sea ice mask M_{GT}, divided by the union area between them, as follows:

$$\text{IoU} = \frac{M_p \cap M_{GT}}{M_p \cup M_{GT}} \tag{1}$$

To assess the accuracy of the segmentation predictions, we used a threshold. If we set the IoU threshold t to be high during the image detection, we could achieve a higher prediction accuracy. In this paper, we varied the IoU thresholds from 0.50 to 0.95, with a step size of 0.05, to obtain an accurate object segmentation.

We counted the detected ice to obtain the number of true positives (*TP*), false positives (*FP*), and false negatives (*FN*). In this study, *TP* refers to the sea ice that the model correctly identified as ice, *FP* refers to the water or other objects (excluding ice) that the model mistakenly identified as ice, and *FN* refers to the sea ice that the model mistakenly identified as water or other objects (excluding ice). We calculated the precision (*P*) and recall (*R*) using the following equations:

$$P(t) = \frac{TP(t)}{TP(t) + FP(t)} \tag{2}$$

$$R(t) = \frac{TP(t)}{TP(t) + FN(t)} = \frac{TP(t)}{GT} \tag{3}$$

where t is the variable used for the different thresholds of IoU. P represents the accuracy of the sea ice predictions, that is, the model's ability to identify sea ice. R represents the percentage of the predicted sea ice among all the sea ice in the image, that is, the model's ability to find all the sea ice. A good instance segmentation model should have both a high precision and high recall. The average IoU and precision of the trained model were calculated to be 90.38% and 89.15%, respectively. Because of the large number of ice blocks in images, the detection speed was lower than the mean speed for detecting other objects with YOLACT, at 1.7 frames per second (FPS).

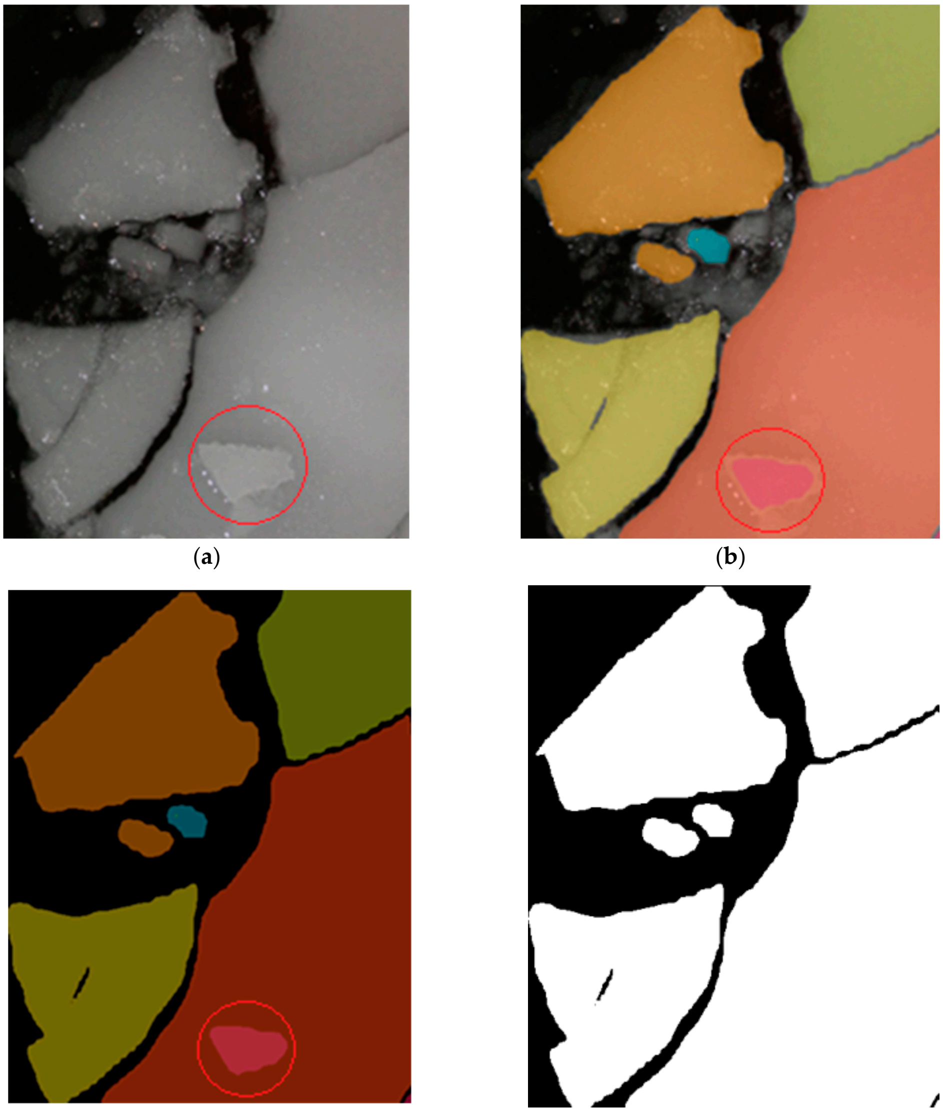
After detecting the ice using the trained model, the mask of the ice blocks could be extracted, which was used to calculate the ice area. The algorithm for the sea ice mask extraction is given in Algorithm 1.

2.4. Calculation of Ice Area

During ice breaking, some crushed ice may overturn and end up on the surface of other floes, as shown in Figure 4b, which was detected from Figure 4a. As the model could detect any ice floes in the image, the floes on top of one another would be detected in the results. When calculating the area of all the masks directly, the overlapped area of the ice floes would cause the ice concentration to be overestimated. In practical applications, when icebreakers are breaking the ice, a lot of ice will overturn, and small pieces of ice will often end up on top of larger floes, making it difficult to accurately calculate the ice concentration.

Algorithm 1. Ice Pixel Extraction

```
Input: sea ice image
Start algorithm:
$WP$← sea ice image
$SEGMENTATION$←an empty list to store the result of instance segmentation
$MODEL$←the trained model from YOALCT
$C$←the confidence threshold
$I$←the IoU threshold
$DETECTION$←the detection result of the sea ice image by using $MODEL$
For each detected mask $m$ in $DETECTION$ do
    If confidence of $m > C$ and IoU of $m > I$ do
        $WP$←$WP$ with $m$
        $SEGMENTATION$←the box, class, and mask
End for
Return $WP$ and $SEGMENTATION$
Output: sea ice segmentation
```



(**a**)

(**b**)

(**c**)

(**d**)

Figure 4. Area correction. (**a**) Original image. (**b**) Instance segmentation of (**a**). (**c**) Extract mask from (**b**). (**d**) Binarization. The red circle indicates overlapped ice.

To solve this problem, we used binarization after the ice floe identification. Binarization is an image processing method that classifies all the pixels of an image according to their grayscale values. Pixels with a grayscale value above a certain threshold are displayed as white and those below the threshold are black. However, the output of the detection is a mask overlaid on the original image, which cannot be directly processed through binarization. To address this, we extracted the mask from the detection result of a new image that consisted of all zero tensors, as shown in Figure 4c. We then converted it into a binarized image, according to the grayscale of each pixel. This allowed us to eliminate the area of an ice floe on top of another, as shown in Figure 4d, and accurately measure the ice concentration by counting the white pixels in this image.

After applying area correction using this method, we calculated the number of pixels belonging to different objects. The number of white pixels represents the area of the ice floes, while the number of black pixels represents the area of the water. Using this information, we collected and labeled all the ice pieces, as shown in Figure 5. To visualize the sizes more effectively, we defined the size of each ice piece as the number of pixels it occupied. We labeled the ice pieces in different colors, based on the following equation:

$$\text{Color}(p) = \begin{cases} 0, & if\ p \notin ICE \\ Colormap([\frac{size(i)}{\max(size(ICE))}]), & if p \in ice(i) \end{cases} \quad (4)$$

where *p* is the pixel, *ICE* = {*ice*(*1*), *ice*(2), *ice*(3), . . . } is a set of detected ice pieces, *Colormap* is an RGB color bar from blue to red, *ice*(*i*)∈*ICE*, *size*(*i*) is the pixel number of *ice*(*i*), and *size*(*ICE*) is the pixel number of each ice piece in *ICE*. The smaller ice pieces are blue and the larger ice pieces are red. The area of the ice was calculated using Algorithm 2. Additionally, we conducted a test using 100 images to calculate the running time of Algorithms 1 and 2, including a sea ice instance segmentation and calculation of the sea ice concentration. Finally, we found that the speed was 21 FPS, which supports the requirements of real-time detection.

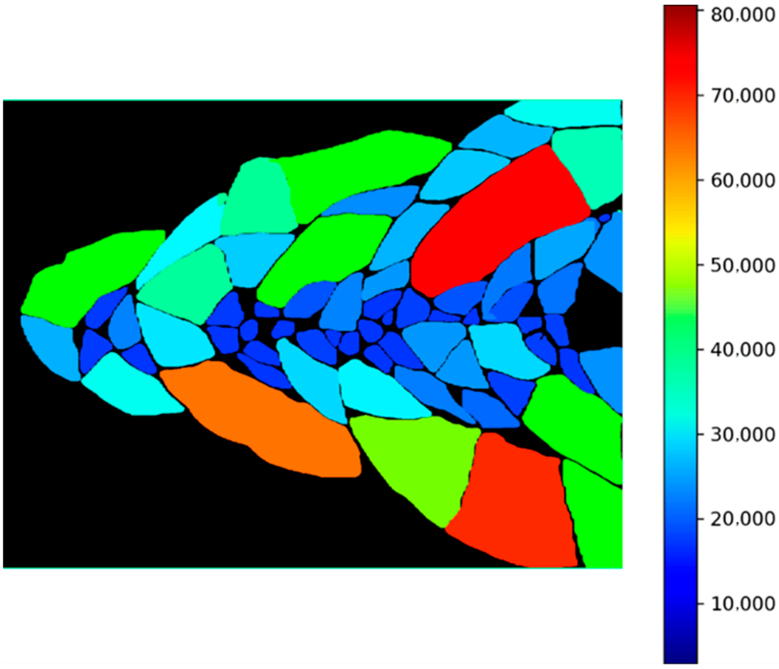


Figure 5. Colorized ice pieces. The color bar means the number of pixels.

Algorithm 2. Ice Area Calculation

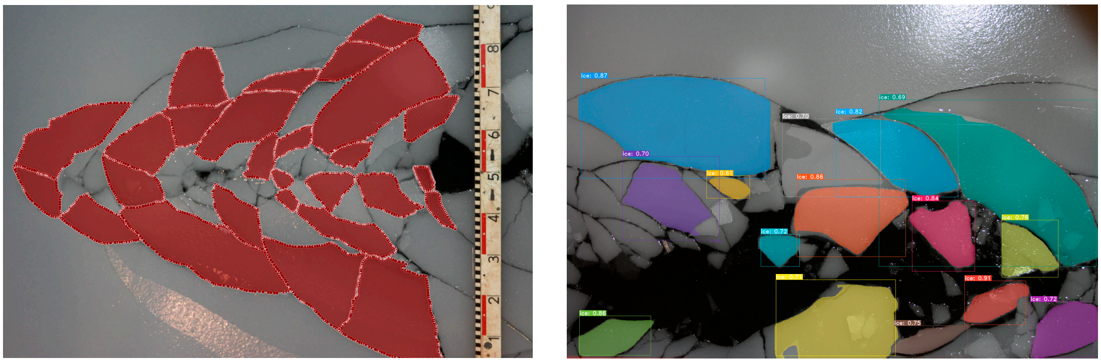
```
Input: ice segmentation
Start algorithm:
PIECES $\leftarrow$ ice segmentation
AREA $\leftarrow$ an empty list to store area of each ice
BW $\leftarrow$ empty black image
for each piece $\in$ PIECES do
    piece $\leftarrow$ binarization of each mask
    AREA $\leftarrow$ area of piece
end for
INDENX $\leftarrow$ the number of sorted AREA
for i $\in$ INDENX do
    COLOR $\leftarrow$ get color by Equation (4)
    pixel of the piece[i] $\leftarrow$ COLOR
  BW $\leftarrow$ BW with colored piece
end for
IDENTIFICATION $\leftarrow$ labeled region in BW
FLOE $\leftarrow$ labeled regions in IDENTIFICATION with large sizes
BRUSH $\leftarrow$ labeled regions in IDENTIFICATION with small sizes
DISTRIBUTION $\leftarrow$ floe size distribution
return FLOE, BRASH, DISTRIBUTION
Output: floe size distribution
```

3. Sensitivity Analysis on Annotation and NMS Parameters

When building a dataset and testing the trained model, we found that the sea ice segmentation could be affected by different annotations in the dataset and different settings of the NMS parameter. In this section, we will elaborate on the reasons for this condition and analyze the sensitivity to different annotations and NMS parameters.

3.1. Annotation of Ice Blocks

When collecting ice images, it is necessary to label the ice blocks in these images to inform the computer what constitutes ice, so that it can identify and learn the characteristics of the ice via deep learning. In a previous study [31], YOLACT was used to detect wedge ice and identify circumferential cracks. However, the last dataset used only labeled some wedge-shaped ice, as depicted in Figure 6a, and did not capture many small ice pieces or the ice around the images. To address this issue, we attempted to use the neural network for identifying the unmarked model ice through a semi-supervised learning approach. Given the large number of ice blocks in each image and the adequate quantity of images, we deemed the dataset to be sufficiently large.



(**a**) (**b**)

Figure 6. (**a**) The label in the first training set. (**b**) The result of ice detection.

However, when using the model to identify the model ice, a significant amount of model ice could not be segmented due to the network characteristics of YOLACT. The convolution layer outputted a feature vector through the full connection layer, which may have had shape features aligned with the model ice boundaries. The detection result was mostly ice that had a similar shape to the labeled ice in the training set, as shown in Figure 6b. Ice blocks of other shapes could not be identified and the mask could not cover the ice completely. In the previous task, detecting wedge-shaped ice and extracting circumferential cracks was sufficient to meet the task requirements. However, the task of this study was to calculate the ice concentration and floe size distribution; therefore, all the ice blocks must be detected, regardless of their shape. Hence, the data set was not suitable for this task.

Based on this experience, we relabeled the model ice to eliminate the above-mentioned issues. We mainly relabeled the ice that was not labeled in the previous labeling process, as shown in Figure 7. In the new data set, we labeled model ice of different shapes to create various training samples. This approach ensured that the model could identify ice of different shapes.

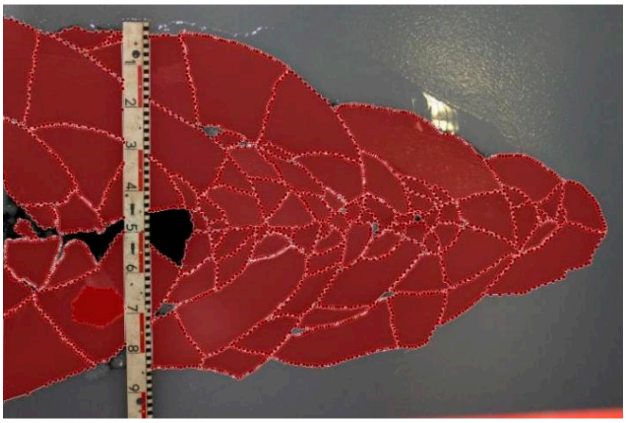


Figure 7. New relabeled data set.

After conducting multiple training tests, we created an instance segmentation dataset specific to sea ice image processing. This dataset could be used to train various models to achieve different functions and the trained model could also be utilized for detecting ice in tanks and open water.

3.2. Recommendation for NMS Parameters

Based on the images captured from the ice tank and open water, ice blocks tend to be in close contact in areas with a high ice concentration. This is in contrast to other object instance segmentation tasks, such as identifying people or cars, where objects generally do not make such close contact. The original YOLACT network can effectively segment the objects in images with generic parameters.

Non-maximum suppression (NMS) acts like a high-pass filter in the YOLACT network, as shown in Figure 1. It filters out boxes with a lower confidence by first sorting all the prediction boxes B in the same class, then selecting the prediction box with the highest score S to calculate the IoU with the other prediction boxes in the same class. It then deletes any boxes larger than the threshold until all the boxes are considered. However, when an object is occluded, as illustrated in Figure 8, the three dashed boxes represent three objects of the same class, while the solid wireframe is a box that frames all three objects together. Despite having the largest IoU threshold M, the NMS selects the solid wireframe with an IoU value of 0.9 as the first box to be retained. If the IoU threshold of the other boxes and this box is greater than the selected threshold, the NMS will delete the three dashed boxes with accurate positioning but low classification scores, resulting in an inaccurate box selection.

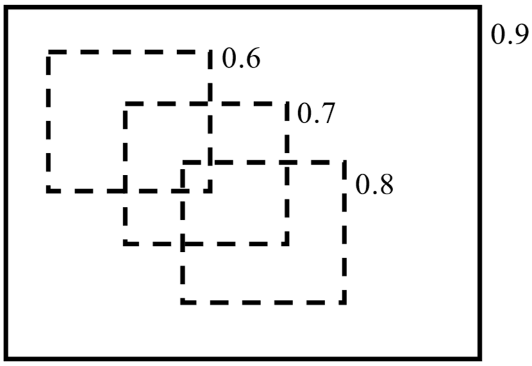


Figure 8. An example of NMS.

As the instance segmentation of ice blocks differs from other object detections and instance segmentations, the NMS threshold determines the number of ice blocks that can be identified. In other words, if two ice blocks highly overlap, they can only be detected by setting a large NMS threshold to ensure that the box with a lower confidence score is not suppressed.

YOLACT proposed a new method called Fast-NMS to enhance the speed of detection. The ROI position can be obtained by adding the position offset to the anchor position. As per the above analysis, NMS is a commonly used filtering algorithm if the ROI overlaps. Matrix simplification is used to reduce the time in Fast-NMS; however, some accuracy has to be sacrificed. For example, let us suppose there are five ice ROIs—B1, B2, B3, B4, and B5, respectively. Next, the IoU between them is obtained through a matrix operation. The hypothetical results are shown in Table 1. In the next step, the lower triangle and diagonal elements of the matrix are deleted to obtain the key parameters presented in Table 2.

Table 1. IoU of 5 ROIs.

	B1	B2	B3	B4	B5
B1	1	0.8	0.1	0.2	0.3
B2	0.8	1	0.6	0.4	0.1
B3	0.1	0.6	1	0.6	0.2
B4	0.2	0.4	0.6	1	0.4
B5	0.3	0.1	0.2	0.4	1

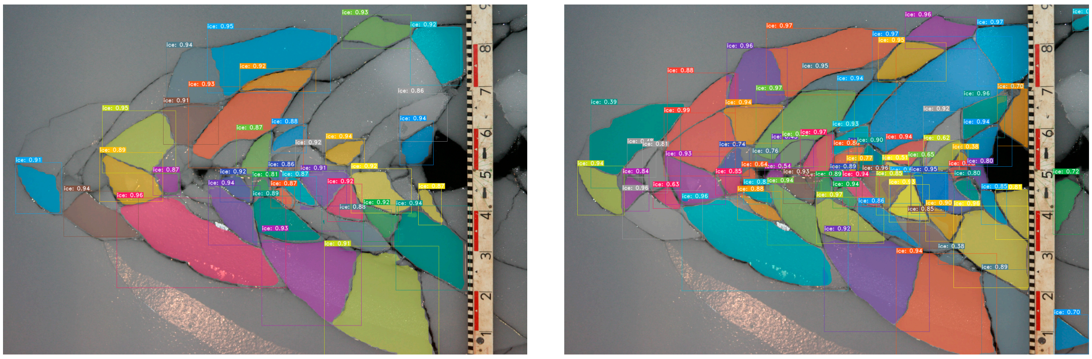
Table 2. Remaining IoUs after removing the small values.

	B1	B2	B3	B4	B5
B1	-	0.8	0.1	0.2	0.3
B2	-	-	0.6	0.4	0.1
B3	-	-	-	0.6	0.2
B4	-	-	-	-	0.4
B5	-	-	-	-	-

Each of these elements satisfies the condition that the row number is less than the column number. Next, the maximum value for each column is taken and the result is (-, 0.8, 0.6, 0.6, and 0.4). Assuming a threshold of 0.5, any two ROIs with an IoU greater than 0.5 should discard the one with a low confidence. According to the maximum value, the columns corresponding to B2, B3, and B4 exceed the threshold; therefore, these three ROIs will be removed in this step.

This is because, as the row number of each element is less than the column number and the sequence numbers are arranged in descending order from high to low confidence, any element greater than the threshold indicates that the ROI corresponding to this column overlaps with the ROI with a higher confidence; therefore, it needs to be removed.

As stated above, B3 is retained because it overlaps too much with B2 (the IoU is 0.6); however, there is a situation where B3 and B2 are the bounding boxes of ice that are closely contacted. There will be some ice that cannot be detected using this method, as shown in Figure 9a. To address this issue, we use traditional NMS instead of Fast-NMS to build a more suitable network for the ice instance segmentation. In the traditional NMS algorithm, we set the number of examples to consider for NMS as 200, which means that the bounding boxes of the top 200 IoU will be considered. Because the number of ice blocks in the image is usually large, a threshold of 0.01 is set to remove the bounding box with a confidence less than the threshold. Additionally, we set another NMS threshold of 0.5 to filter out the boxes with an IoU overlap less than this threshold. With these parameters, the algorithm can identify the ice in the ice tank well, as shown in Figure 9b.



(**a**) (**b**)

Figure 9. Different result of the same image. (**a**) The result of higher threshold of NMS. (**b**) The result of the threshold as above setting.

4. Case Study

In this section, we will select an image captured on an icebreaker sailing through a cold region, process this image using the methods introduced in the previous section, and calculate the sea ice concentration and floe distribution.

4.1. Extract the Mask of Ice

Usually, the camera is placed at the bow of the ship and shoots in the forward direction of the ship at an angle of approximately 45°, as shown in Figure 10. The sea ice detection algorithm is applied to the image to obtain the mask of the sea ice, which is then converted into a binary format. The number of pixels per mask is calculated to determine the size of the ice floe, and the sum of the pixels for all the masks represents the area of the ice floe in the region that is measured.

When measuring the sea ice concentration in the bow region, the camera must be positioned at a certain angle to capture an image that includes the bow area. To accurately calculate the sea ice concentration, the pixels corresponding to the bow area in the image need to be removed. Additionally, the model must be able to identify different types of ships in the image. To accomplish this, we augment the dataset by adding the training data from ships sailing in cold regions and combine them with the original dataset. The pretraining model we use is based on the model ice. A new model is then trained, which can identify both ships and ice simultaneously.

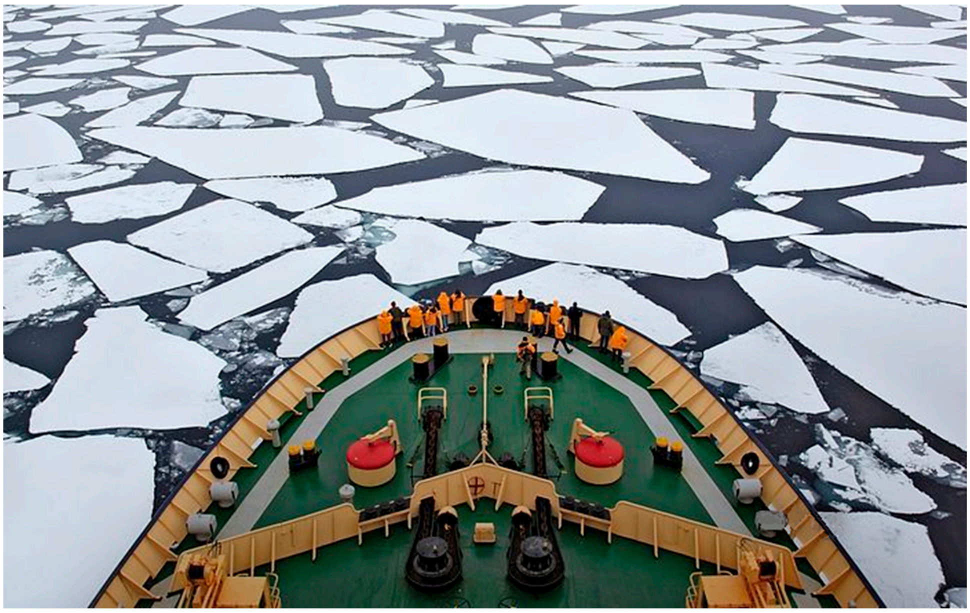


Figure 10. Image taken with the ship in Antarctica [32].

Figure 10 was detected with the new model, as shown in Figure 11. The sea ice in the bow region is accurately identified. However, due to the shooting angle, the mask of the sea ice detected in the distance is incomplete. The shooting angle not only affects the detection accuracy, but also causes the amount of sea ice detected in the distance to be less than its actual value. In practice, it is difficult to avoid the shooting angle, as this is necessary to ensure that the image contains a large enough area.

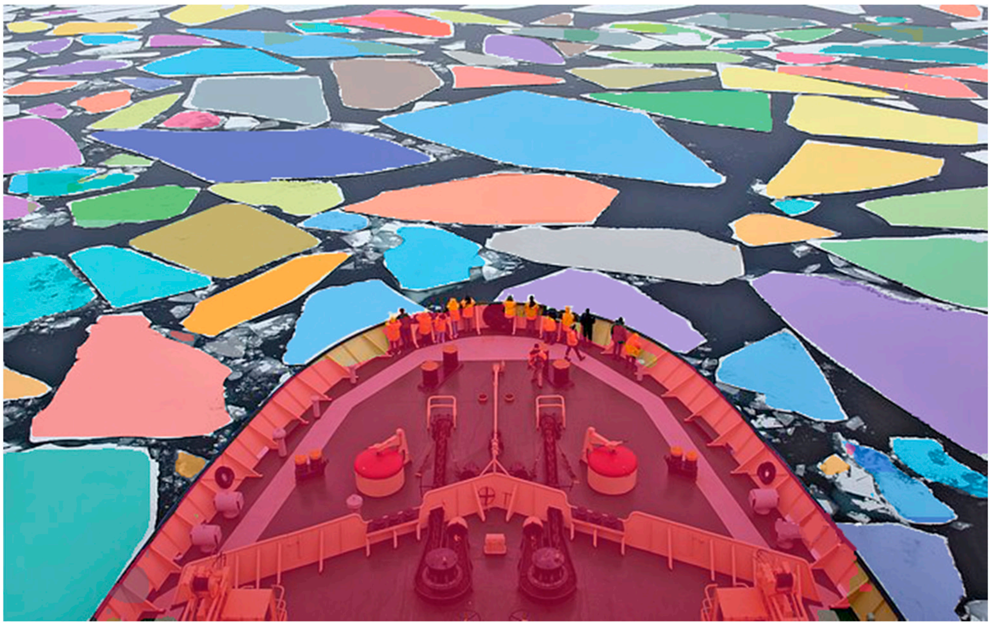


Figure 11. Instance segmentation of Figure 10.

4.2. Geometric Calibration

When sailing in polar regions, navigation observations are captured by camera lenses with outward-looking orientations. These cameras take pictures of the bow and shoulder areas; however, to capture a larger area of sea ice around the ship, they are set at a wide angle instead of vertical shooting. However, this can result in inadequate identification results for calculating the size distribution statistics. The calculated area of the ice floes in the far range of the image may be smaller than the actual area, introducing errors into further analyses.

According to [15,16], the field of view and shooting angle of the camera can be measured by a sensor and this distortion can be orthorectified. However, in Figure 10, the actual parameters of the camera are not measured. Therefore, we estimate the shooting angle and field of view based on the statistical similarities between the size distributions of the near and far ranges of the image. In Figure 10, the shooting angle is approximately 45° and the field of view is 46°. Using this information, we can orthorectify the overall segmentation image and illustrate the algorithm specifically.

As shown in Figure 11, some ice floes at the far range of the image may not be identified due to distortion. Therefore, a geometric calibration should be performed on the original sea ice image before the ice instance segmentation (Algorithm 1). In this paper, a perspective transformation is used for the geometric calibration, which is a form of projective mapping that projects the picture onto a new viewing plane. Perspective transformation is used to transform objects into straight lines from an original picture, as objects that are straight lines may appear as slashes in the picture. The picture is transformed from the focal plane to the orthorectification plane through a linear transformation and translation, with the transformation equation being as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{5}$$

where (X, Y, Z) is the coordinate of the focal plane and the transformation matrix is: $\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$.

The transformation matrix can be divided into four parts: $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$, which denotes the linear transformation, [a_{31} a_{32}], which denotes the translation, and $[a_{13}\ a_{23}]^{\mathrm{T}}$, which leads the perspective. The coordinate of the corresponding point in the transformed picture is (X', Y', Z'). Because we are processing a two-dimensional image, we assume Z' = 1 and divide the transformed image coordinates by Z', reducing the image from three dimensions to two dimensions. Then, we can obtain the following equation:

$$\begin{cases} X' = \frac{X}{Z} \\ Y' = \frac{Y}{Z} \\ Z' = \frac{Z}{Z} \end{cases} \tag{6}$$

$$\begin{cases} X' = \frac{a_{11}x+a_{12}y+a_{13}}{a_{31}x+a_{32}y+a_{33}} \\ Y' = \frac{a_{21}x+a_{22}y+a_{23}}{a_{31}x+a_{32}y+a_{33}} \\ Z' = 1 \end{cases} \tag{7}$$

Generally, to calculate X' and Y', we assume a_{33} = 1, which yields:

$$\begin{cases} a_{11}x + a_{12}y + a_{13} - a_{31}xX' - a_{32}X'y = X' \\ a_{21}x + a_{22}y + a_{23} - a_{31}xY' - a_{32}yY' = Y' \end{cases} \tag{8}$$

Therefore, given the several points corresponding to the transformation, the transformation equation can be obtained. We choose four points of the original picture, such as (x_0,y_0), (x_1,y_1), (x_2,y_2), and (x_3,y_3). Furthermore, we set four target points as the vertices of the transformed picture: $(X_0{}',Y_0{}')$, $(X_1{}',Y_1{}')$, $(X_2{}',Y_2{}')$, and $(X_3{}',Y_3{}')$. By substituting these points into Equation (8), we can obtain:

$$\begin{bmatrix} x_0 & y_0 & 1 & 0 & 0 & 0 & -x_0X_0' & -y_0X_0' \\ 0 & 0 & 0 & x_0 & y_0 & 1 & -x_0Y_0' & -y_0Y_0' \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1X_1' & -y_1X_1' \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1Y_1' & -y_1Y_1' \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2X_2' & -y_2X_2' \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2Y_2' & -y_2Y_2' \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3X_3' & -y_3X_3' \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3Y_3' & -y_3Y_3' \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \end{bmatrix} = \begin{bmatrix} X_0' \\ Y_0' \\ X_1' \\ Y_1' \\ X_2' \\ Y_2' \\ X_3' \\ Y_3' \end{bmatrix} \tag{9}$$

The relationship between (x, y) and (X, Y) can be found on the basis of [16]. By substituting the relation equations into (9), the transformed matrix can be obtained.

Once we obtain the transformation matrix, we can perform the perspective transformation on every pixel in the image using Equation (5), as shown in Figure 12.

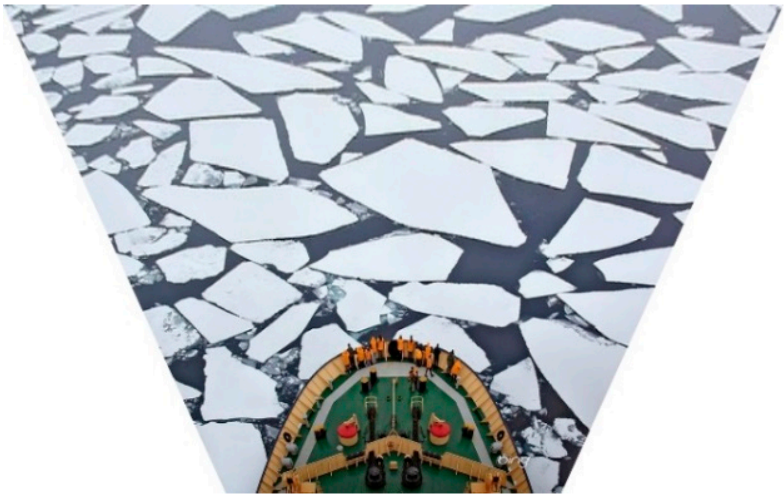


Figure 12. Result after orthorectification.

If we compare Figures 10 and 12, it is clear that the perspective transformation transforms the distorted image into a ″bird′s-eye view″ of the ice field, as illustrated in Figure 13. The view of the ice field has changed from a perspective shot from the ship to a vertical shot over the ice field. The ice in the far range of the picture has been enlarged, while the ice in the close range of the picture has been reduced.

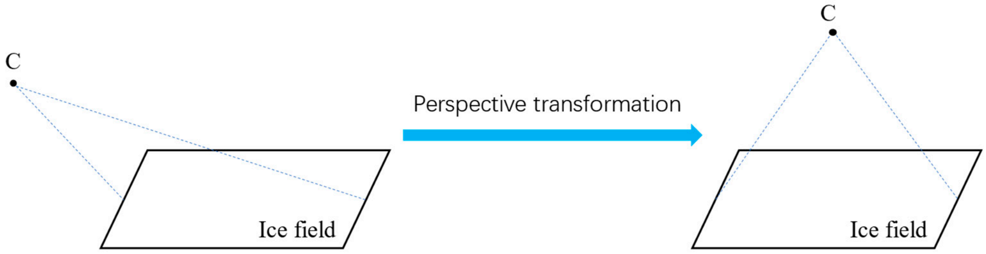


Figure 13. Change of view after perspective transformation.

4.3. Sea Ice Concentration and Floe Size Distribution

According to [16], a brash ice threshold parameter is used to distinguish brash ice from ice floes based on the area of the identified ice. Brash ice is considered to be floating ice fragments no more than 2 m across; thus, ice pieces larger than the threshold are considered to be ice floes, while smaller pieces are considered to be brash ice.

After inputting the picture following the perspective transformation, we obtain the ice floe and brash ice size distributions, as shown in Figure 14. The color of the ice ranges from blue to red according to the area of the ice blocks, with the brash ice shown in dark blue, smaller floes in light blue, and larger floes in red.

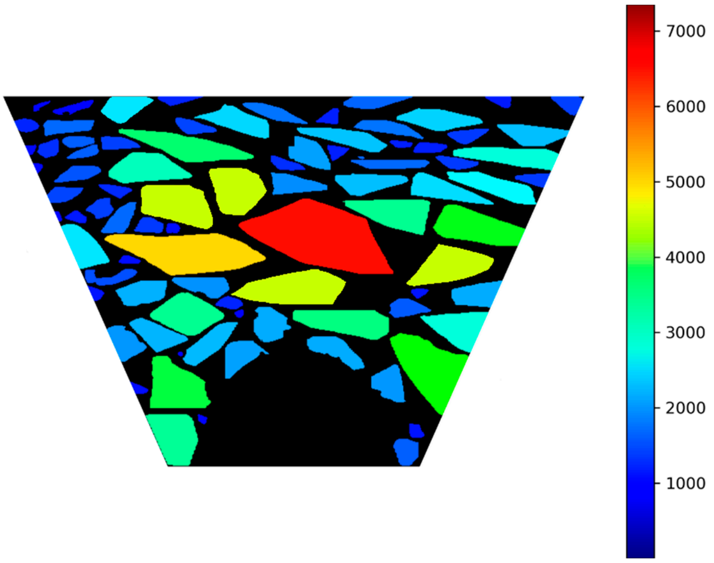


Figure 14. Ice floe distribution after orthorectification. The color bar means the number of pixels.

Using the masks of the ice blocks, we can determine the pixel count of the mask and calculate the ice concentration using the equation:

$$concentration = \frac{\sum area(m)}{area(P)} \qquad m \in MASK \tag{10}$$

where $MASK = \{m(1), m(2), m(3), \ldots\}$ is a set of detected masks and $area(P)$ is the pixel number of the whole picture. However, after the perspective transformation is applied to the original picture, the size of the picture changes. Therefore, we calculate the pixel number of Figure 14 as the new $area(P)$. In Figure 14, there are 87 ice floes and 1 ship identified. The percentages of coverage are 9.34% of brash ice, 45.12% of ice floes, and 45.53% of water. As the height of shooting the picture is unknown, the actual size of the ice can be calculated based on the number of pixels in the detected sea ice. The ice-floe-distributing histogram is yielded and shown in Figure 15.

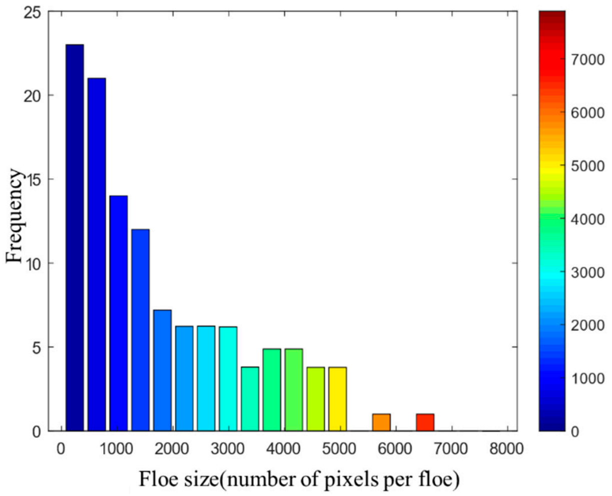


Figure 15. Ice floe size distribution histogram of Figure 14.

5. Discussion

The method used in this paper took into account the different objects and illuminations in images. Instance segmentation was first used to calculate the ice concentration and size distribution. Shipboard oblique photography is an unavoidable challenge when measuring sea ice. Although possible collisions with ice floes during ice breaking may affect the ice definition, our method is still suitable for regions with low ice concentrations, such as the marginal ice zone. Collisions only occur in regions with heavy ice conditions.

Due to the reflection of light sources on the water and ice, the bright characteristics of light may cause water to be identified as ice floes with threshold segmentation. To train a suitable instance segmentation model to accurately detect sea ice, a dataset of this sea ice was constructed. The dataset contained images of model ice in the ice tank and images of ships sailing through cold regions with varying lighting conditions, allowing the computer to accurately identify the real sea ice. All shapes of the ice and ships in the images were labeled, enabling the model to learn sufficient features. The instance segmentation model trained with this dataset can meet many working conditions in the process of analyzing sea ice images.

6. Conclusions

Compared to remote sensing, images obtained from navigation observations have a higher resolution. Various image processing methods can be applied to these images to obtain important sea ice information, such as sea ice concentration and size distribution in the forward area, which can improve ship handling safety.

To identify the sea ice from images containing both ice and ships and derive the ice concentration, we proposed an algorithm using instance segmentation based on the YOLACT network. In the network, the NMS threshold was adjusted to detect the closely connected ice in high-concentration broken ice fields. The masks outputted from YOLACT were binary, in order to remove the masks of ice blocks that were covered by others, so that the accuracy of the floe size could be calculated. We used perspective transformation in a geometric orthorectification to correct for the situation where ice blocks appeared big when near and small when far and obtained an acceptable sea ice concentration.

The developed methods provided valuable ice information that can be used to improve our understanding of the sea ice in polar regions and navigation safety. However, it is necessary to collect more image data to extend the dataset and quantitatively validate the present method of calculating ice concentration. Images from different working conditions are also needed to provide a basis for comparison with this study and this will be addressed in future research.

Author Contributions: Resources and project administration, L.Z.; software and writing original draft preparation, J.C.; validation, S.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Key Technologies Research and Development Program, grant number 2022YFE0107000 and National Natural Science Foundation of China, grant number 52171259.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions.

Acknowledgments: We would like to thank the anonymous reviewers for their constructive and valuable suggestions on the earlier drafts of this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, L.; Riska, K.; Ji, C.Y. Simulating transverse icebreaking process considering both crushing and bending failures. *Mar. Struct.* **2017**, *54*, 167–187. [CrossRef]
2. Wadhams, P. *Ice in the Ocean*; CRC Press: Boca Raton, FL, USA, 2000.

3. Keinonen, A.J. Ice Management for Floating Ice Offshore Operations. In Proceedings of the Offshore Technology Conference, Houston, TX, USA, 5–8 May 2008; pp. 690–704.
4. Hamilton, J.; Holub, C.; Blunt, J.; Mitchell, D.; Kokkinis, T. Ice management for support of arctic floating operations. In Proceedings of the OTC Arctic Technology Conference, Houston, TX, USA, 7–9 February 2011; pp. 615–626.
5. Jin, Q.; Zhang, J.N.; Zhang, L.; Tian, Y.K. Research on the construction of ship's numerical brash ice field and calculation method of ice resistance. *J. Ship Mech.* **2021**, *25*, 879–890.
6. Xie, C.; Zhou, L.; Ding, S.F.; Liu, R.W.; Zheng, S.J. Experimental and numerical investigation on self-propulsion performance of polar merchant ship in brash ice channel. *Ocean. Eng.* **2023**, *269*, 113424. [CrossRef]
7. Kim, J.H.; Kim, Y. Numerical simulation on the ice-induced fatigue damage of ship structural members in broken ice fields. *Mar. Struct.* **2019**, *66*, 83–105. [CrossRef]
8. Jeong, S.Y.; Jang, J.; Kim, C.H.; Yum, J.G.; Kang, K.J. A study of ship resistance characteristics for ice-strengthened vessel by broken ice channel width and size of broken ice pieces. *J. Soc. Nav. Archit. Korea* **2018**, *55*, 22–27. [CrossRef]
9. Luo, W.Z.; Jiang, D.P.; Wu, T.C.; Guo, C.Y.; Wang, C.; Deng, R.; Dai, S.S. Numerical simulation of an ice-strengthened bulk carrier in brash ice channel. *Ocean. Eng.* **2020**, *196*, 106830. [CrossRef]
10. Xue, Y.Z.; Liu, R.W.; Li, Z.; Han, D.F. A review for numerical simulation methods of ship–ice interaction. *Ocean. Eng.* **2020**, *215*, 107853. [CrossRef]
11. Wang, C.; Hu, X.H.; Tian, T.P.; Guo, C.Y.; Wang, C.H. Numerical simulation of ice loads on a ship in broken ice fields using an elastic ice model. *Int. J. Nav. Archit. Ocean. Eng.* **2020**, *12*, 414–427. [CrossRef]
12. Haugen, J.; Imsland, L.; Løset, S.; Skjetne, R. Ice observer system for ice management operations. In Proceedings of the Twenty-First International Offshore and Polar Engineering Conference, Maui, HI, USA, 19–24 June 2011; p. 1120.
13. Lu, P.; Li, Z.J.; Zhang, Z.H.; Dong, X.L. Aerial observations of floe size distribution in the marginal ice zone of summer Prydz Bay. *J. Geophys. Res. Ocean.* **2008**, *113*, C02011. [CrossRef]
14. Zhang, Q.; Skjetne, R.; Løset, S.; Marchenko, A. Digital image processing for sea ice observations in support to Arctic DP operations. In Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, Rio de Janeiro, Brazil, 1–6 July 2012; Volume 44939, pp. 555–561.
15. Lu, P.; Li, Z.J. A method of obtaining ice concentration and floe size from shipboard oblique sea ice images. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 2771–2780.
16. Zhang, Q.; Skjetne, R. Image processing for identification of sea ice floes and the floe size distributions. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 2913–2924. [CrossRef]
17. Makynen, M.P.; Cheng, B.; Simila, M.H.; Vihma, T.; Hallikainen, M.T. Comparisons between SAR backscattering coefficient and results of a thermodynamic snow/ice model for the Baltic Sea land-fast sea ice. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 1131–1141. [CrossRef]
18. Kurtz, N.T.; Markus, T.; Cavalieri, D.J.; Krabill, W.; Sonntag, J.G.; Miller, J. Comparison of ICESat data with airborne laser altimeter measurements over Arctic sea ice. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 1913–1924. [CrossRef]
19. Yu, P.; Clausi, D.A.; Howell, S.E. Fusing AMSR-E and QuikSCAT imagery for improved sea ice recognition. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 1980–1989. [CrossRef]
20. Awange, J.L.; Kiema, J.B.K. Microwave remote sensing. In *Environmental Geoinformatics*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 133–144.
21. Inoue, J.; Curry, J.A.; Maslanik, J.A. Application of Aerosondes to melt-pond observations over Arctic sea ice. *J. Atmos. Ocean. Technol.* **2008**, *25*, 327–334. [CrossRef]
22. Yan, Q.Y.; Huang, W.M. Sea ice sensing from GNSS-R data using convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1510–1514. [CrossRef]
23. Zhu, Y.; Tao, T.; Li, J.; Yu, K.; Wang, L.; Qu, X.; Li, S.; Semmling, M.; Wickert, J. Spaceborne GNSS-R for Sea Ice Classification Using Machine Learning Classifiers. *Remote Sens.* **2021**, *13*, 4577. [CrossRef]
24. Worby, A.P.; Allison, I.; Dirita, V. *A Technique for Making Ship-Based Observations of Antarctic Sea Ice Thickness and Characteristics*; Antarctic CRC: Boca Raton, FL, USA, 1999.
25. Lu, P.; Li, Z.J.; Dong, X.L.; Zhang, Z.H.; Chen, Z. Sea ice thickness and concentration in Arctic obtaining from remote sensing images. *Chin. J. Polar Sci.* **2004**, *15*, 91–97.
26. Zhang, Q.; van der Werff, S.; Metrikin, I.; Løset, S.; Skjetne, R. Image processing for the analysis of an evolving broken-ice field in model testing. In Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, Rio de Janeiro, Brazil, 1–6 July 2012; Volume 44939, pp. 597–605.
27. Lu, W.J.; Zhang, Q.; Lubbad, R.; Løset, S.; Skjetne, R. A Shipborne Measurement System to Acquire Sea Ice Thickness and Concentration at Engineering Scale. *Polar Remote Sens.* **2016**, *1*, 1–12.
28. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
29. Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. YOLACT++: Better Real-time Instance Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1108–1121. [CrossRef]

30. Zhao, Q.J.; Sheng, T.; Wang, Y.T.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H.B. M2det: A single-shot object detector based on multi-level feature pyramid network. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 29–31 January 2019; Volume 33, pp. 9259–9266.
31. Cai, J.Y.; Ding, S.F.; Zhang, Q.; Liu, R.W.; Zeng, D.H.; Zhou, L. Broken ice circumferential crack estimation via image techniques. *Ocean. Eng.* **2022**, *259*, 111735. [CrossRef]
32. Available online: https://www.pinterest.com.au/pin/310959549242558457/ (accessed on 12 November 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.