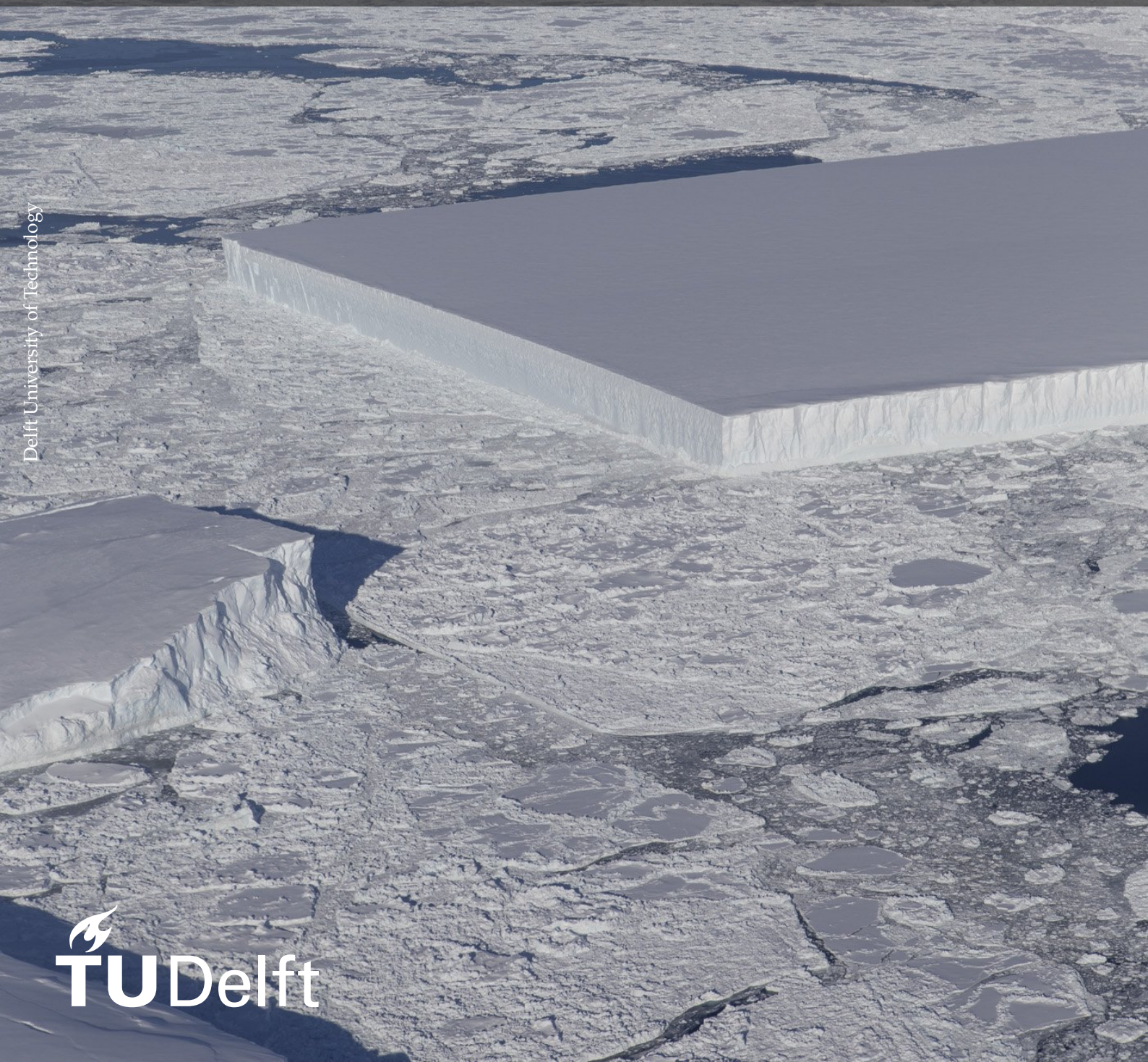
Using YOLOv5 for the Detection of Icebergs in SAR Imagery

Daan Hulskemper



Delft University of Technology

TU Delft

Using YOLOv5 for the Detection of Icebergs in SAR Imagery

by

Daan Hulskemper

Student Name	Student Number
Daan Hulskemper	5183227

Supervisor:	S. Lhermitte
Second Supervisor:	R. Taormina
Project Duration:	February, 2022 - May, 2022
Faculty:	Faculty of Civil Engineering and Geosciences, Delft

Cover: A tabular iceberg floating among sea ice just off of the Larsen C ice shelf by NASA

TU Delft

Abstract

This research aims to analyse the sensitivity of the YOLOv5 object detection algorithm to current issues related to the tracking of icebergs in SAR imagery. To this end a sensitivity study was done on (1) the sensitivity of the algorithm to variations in input image resolution, (2) the sensitivity of the algorithm to variations in contrast between an iceberg and its surroundings and (3) the sensitivity of the algorithm to variations in icebergs size. The results show that the algorithm is very robust against variations in contrast between iceberg and surroundings, but is significantly sensitive to iceberg size. Furthermore, it seems that only by using high resolution images, the spatial features of icebergs can be well distinguished from features of other objects in the ocean. The YOLOv5 algorithm thus shows great potential for iceberg detection applications, but it should be explored if the current sensitivity to size can be overcome if a more evenly distributed training dataset is used. On top of this, it should be noted that this research only serves as an exploratory analysis on the application of the algorithm and it should thus still be explored if our results based on augmented data, also apply on real data.

Summary

The detection and tracking of icebergs in the polar regions is a significant field of study, as icebergs pose potential hazards for ship navigation, alter oceanic currents and sea-ice formation and influence the biological and chemical state of the ocean. Their locations and trajectories also contain information on oceanic currents and weather patterns. Current iceberg detection and tracking is either done manually, which is a very time-intensive work, or (semi)automatically. The (semi)automated methods in general focus on using thresholding and segmentation methods on SAR imagery. These methods however suffer from the fact that icebergs and its surroundings can show irregular variations in contrast in the backscatter imagery, due to melting on top of the iceberg, variations in atmospheric and oceanic conditions (sea-ice, rough weather and snow) and the nature of SAR imagery. Furthermore, the large variations in iceberg size, especially in the Antarctic ocean, also poses problems in applying a detection and tracking method on a global scale.

The YOLOv5 object detection algorithm could overcome these current issues with automated detection, as this deep learning network contains a large amount of trainable feature extractors, should be largely scale invariant and is not fixed to a certain input image resolution. This study therefore explores the applicability of YOLOv5 on the automated detection of icebergs in SAR imagery. To this end we trained the YOLOv5 model on a dataset of Sentinel-1 imagery containing tabular icebergs around Antarctica, as tracked by the USNIC. We performed a sensitivity analysis on three topics: (1) the sensitivity of the algorithm to variations in input image resolution, (2) the sensitivity of the algorithm to variations in contrast between an iceberg and its surroundings and (3) the sensitivity of the algorithm to variations in icebergs size.

The first analysis was done by training three models with input sizes of 640x640, 1600x1600 and 3200x3200 pixels. It was found that the 3200 model performed significantly better than the others. Using higher resolutions the algorithm can clearly distinguish spatial features of icebergs from features of other oceanic objects. The second analysis was done by pasting four snipped out icebergs on a variety of oceanic backgrounds, resulting in varying contrasts. Here in negative, positive and even low contrast situations icebergs were detected. This thus shows that the algorithm is indeed very robust against the variety of contrast situations in which an iceberg can occur in SAR imagery, which gives greater prospect for large scale applications compared to current thresholding methods. The third analysis was done by inflating and deflating the snipped out icebergs and pasting them on one background. It was found that icebergs were detected only when their size fell inside a dense cluster of sizes appearing in the training data. The YOLOv5 algorithm is thus, in contrast to its proposed scale invariance, very sensitive to size. It could however be that when the size of the icebergs in the training data is more equally distributed, the model will perform well on all iceberg sizes.

The results thus show that the YOLOv5 algorithm could potentially be applied to overcome the current issues with contrast variations in SAR imagery, but its potential robustness against scale should be further explored by training on a more equally distributed dataset. Furthermore, it should be explored how our results extrapolate to real test data.

Contents

1
Introduction

1.1. Characteristics and impact of icebergs

Icebergs form when parts of an ice sheet or glacier that terminate in the ocean, the ice shelves or glacier mouths, break off. This calving takes place at the locations of crevasses, which are formed once stresses on the shelf exceed a certain strength[8]. The size of these icebergs ranges from less than a square kilometer in area up to several thousands of square kilometers[31]. Around Antarctica these icebergs mainly originate from large flat ice sheets, consequently the icebergs have a table like form, and are called tabular icebergs. Around the Arctic circle this tabular form is less apparent and icebergs in general are smaller.

The trajectories of icebergs influence and are influenced by the surrounding ocean and atmospheric state. First of all, once an iceberg is detached it can travel the ocean for decades before disintegrating completely through consecutive break ups and basal melting. During this time its trajectory is influenced by oceanic currents, weather patterns and sea-ice patterns[22, 14], while the icebergs itself also influences these patterns. As such knowledge of the locations and trajectories of icebergs contains valuable information on the state of these interactions. Icebergs also serve as a large, cold and mineral rich, fresh water source, as such the presence of icebergs can locally and globally influence oceanic currents[24, 25], sea-ice formation[18] and biological and chemical processes in the ocean[15, 23]. Changes in the amount and magnitude of calving events also influence and are influenced by the earth's changing climate, as the detachment of icebergs impact the mass balance of the ice sheets. Furthermore, the presence of icebergs serves as a large potential hazard for ship navigation in the polar regions. Knowledge of the locations and trajectories of icebergs therefor is very valuable and obtaining these is a significant field of study.

1.2. Iceberg detection and tracking

Icebergs are currently detected and tracked both manually and (semi)automatically. The United States National Ice Center (USNIC) is one of the institutes that does this manually. They keep a database of the locations and sizes of all the tabular icebergs around Antarctica with an area larger than 20 square nautical miles (sqNM), or a major axis of more than 10 nautical miles (NM). To this end, they inspect multispectral, thermal and SAR satellite imagery to localize and quantify these icebergs[29]. This is a very time intensive work and therefore only large icebergs are monitored. In order to localize and track the large variety of iceberg sizes on a global scale automated methods are thus needed.

Several automated methods for iceberg detection and tracking already exist but these are not yet applied on a large scale. The current methods are largely based on the analysis of either SAR or scatterometer imagery. This is imagery originating from active sensors that send out beams of electromagnetic waves in the microwave part of the spectrum (300 MHz - 300 GHz) and measure the amount of power received back (backscatter). The amount of backscatter is a function of the geometry and dielectric properties of the incident surface.

SAR and scatterometer imagery hold several great advantages for the automated detection and tracking of icebergs. First of all, clouds transmit radiation in this part of the spectrum and the resulting image is thus not significantly affected by them, especially in the cloudy polar regions this is a large

advantage. The fact that the sensor is active also means that the satellite is not reliant on daylight, and one can thus obtain imagery even in the long polar nights. Furthermore, in contrast to the visible and infrared part of the spectrum, icebergs do show distinctive reflectance characteristics in the microwave part of the spectrum and as such they can be more easily distinguished from the surrounding area[4]. Oceans in general, have a weaker backscatter signal than sea-ice, which has a weaker amount of backscatter than (snow-covered) land and icebergs. Icebergs show a great amount of backscatter, due to their large amount of volume scattering. Thus making them in general distinguishable from sea-ice and open ocean. Furthermore, because of the spatial homogeneity of the backscatter signal of an iceberg, one can detect icebergs by identifying areas with large (larger than a certain threshold) and a spatially homogeneous amount of backscatter (i.e. a combination of thresholding and segmentation)[4, 27].

Several studies have been done that try to utilize these properties (semi)automatically. Another large database of iceberg locations from the Brigham Young University (BYU), for example is obtained by using imagery from scatterometers and identifying homogeneous high backscatter regions[16]. Here however, the imagery is still manually inspected afterwards to define the actual icebergs and work is thus time intensive. In contrast, Stephen and Long (2002) showed that it was possible to detect and track an iceberg semi-automatically by making use of the cross correlation between separate scatterometer images[26]. Automated methods of iceberg detection and tracking with SAR imagery have also been developed in various examples, with fairly accurate results[20, 3, 12, 17, 13, 19, 2]. SAR imagery yields a higher resolution, which means that it is better capable of distinguishing smaller icebergs and iceberg areas can be more accurately estimated.

Most of these methods rely on setting an, in some cases adaptive, parameter based on knowledge or estimation of the difference in backscatter between an iceberg and its surroundings, be it sea-ice, open water, or land. Segments with a backscatter larger than a certain threshold that also exceed a certain homogeneity or area threshold, are then classified as icebergs[12, 13, 19, 2]. The performance of these methods thus is very sensitive to changes in the contrast between iceberg and surroundings. Karvonen et al. (2021) for example were capable of accurately tracking an iceberg in open water, but they note that their method has to be adapted if applied on icebergs surrounded by sea-ice. Furthermore, most of these methods are not capable of detecting and tracking icebergs outside of a defined range of sizes. These factors pose large constraints when applying the methods on a large scale.

1.3. Issues with current methods of automated iceberg tracking

The current automated detection and tracking methods thus suffer from the fact that icebergs and its surroundings in SAR imagery can show a varying contrast from image to image. These variations arise due to the characteristics of icebergs, the oceanic and atmospheric conditions and the nature of remote sensing imagery. First of all, the backscatter signal of the iceberg itself varies when an iceberg partially melts and wet snow or pools appear, as occurs often in summer[26]. The backscatter of the iceberg is then reduced by a large portion due to the low amount of volumetric scattering of water; the iceberg will thus seemingly merge with the background (upper right in Figure 1.1). Secondly, a change in contrast between the iceberg and surrounding ocean can occur due to rough atmospheric conditions (snowfall and wind) or the presence of snow on sea-ice with an equal backscatter as icebergs[30, 19](lower left in Figure 1.1). A third contrast variation occurs due to the nature of the backscatter measurements in SAR (and radar) imagery (lower right in Figure 1.1)[17]. Here the angle of incidence of the beam can cause variations in the amount of backscatter independent of the physical characteristics of the medium. This can also cause a lack of spatial homogeneity inside the iceberg as is visible in the lower right image of Figure 1.1.

These issues with SAR imagery of icebergs impose that a perfect iceberg detection and tracking algorithm should be robust against varying contrast conditions, and thresholding techniques are thus not always applicable. Koo et al. (2021) and Barbat et al.(2021) showed that by relying less on specific backscatter contrasts (thresholding) and more on shape characteristics of the iceberg in consecutive iceberg matching, an algorithm can be obtained that is more robust against contrast variations. Still, these methods rely on predefined thresholds to detect an iceberg in the first place before matching it to previous images. It is thus not clear if melting icebergs, with low backscatter or icebergs in other of the previously mentioned low contrast situations, can in every case be tracked. A method should thus be developed that is fully shape based and/or capable of using various contrast thresholds at once to identify icebergs.

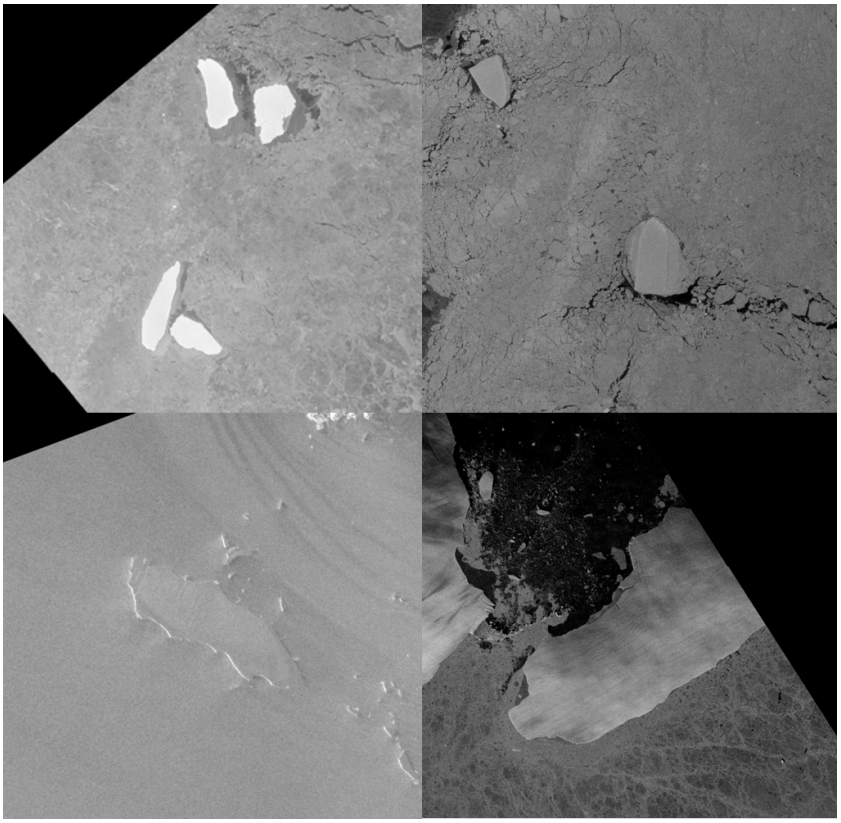


Figure 1.1: Varying spectral and geometric characteristics of icebergs in SAR imagery. Clockwise from the top left: four very clear icebergs close together, with matching geometric shapes; two dark icebergs in a low contrast situation, due to melting of the surface; a partially cutoff iceberg, with varying brightness, likely due to the incidence angle of the scan; a very low contrast image, due to either a rough ocean surface or snow on top of both sea-ice and iceberg. Source: ESA Sentinel-1 imagery.

1.4. Object detection and tracking with deep learning

Deep learning algorithms for object detection and tracking could overcome these current issues related to the detection and tracking of icebergs. These in general do not need preset thresholding and could be trained on a large variety of icebergs, which can make them robust against a large variation of iceberg shape and contrast signatures. The methods are already proven to be very useful in a range of applications[9, 1, 32], with many new deep learning object detection and tracking networks still being made and improved. The networks that are used rely on at first a detection and classification of an object using different layers of convolutional neural networks (CNN). Here, automated feature training and feature extraction of objects in an image is done. The methods thus do not need manual identification or computation of features based on knowledge of the physical state of the object and surroundings. If an appropriate training set is used, varying spectral and spatial signatures of objects can still be classified as the same object class. This proves to be a large advantage when used on objects that change appearance in between images as a result of f.e. occlusion of the object[32]. As icebergs are also frequently changing shape and appearance, these networks might be useful in solving the current problems related to iceberg detection and tracking.

YOLOv5 (*You-Only-Look-Once*) is one of the deep learning object detection algorithms that might be capable of overcoming the current issues with iceberg detection. This algorithm does, in contrast with most object detection algorithms, not have a fixed input image resolution and as such it can be explored what image resolution is needed to define spatial features that distinguish icebergs from other objects in the polar oceans[11]. The algorithm should also be largely scale invariant, as it contains three feature extractors with varying sizes. The performance could thus in theory be robust against the large variety of iceberg sizes that exist. Furthermore, the network contains many layers of features to be trained, relating to the different shapes and contrast situations in which an iceberg can occur. Therefore it can possibly identify icebergs under a range of different shape and contrast circumstances, thus making it potentially robust against the main issues occurring with iceberg detection in SAR imagery. The

algorithm however only carries out object detection and hence the potential of tracking icebergs with deep learning algorithms cannot yet be explored by using it. YOLOv5 can however be used as an input for the DeepSORT tracking algorithm and combining them could thus give a completely automated iceberg tracking workflow in the future.

1.5. Goals and research question

The overarching goal of this research is to give a preliminary insight in the potential of the YOLOv5 algorithm to overcome the current issues with the detection of icebergs in SAR imagery. In order to do so it has to be explored if the algorithm is indeed robust against the varying contrast situations in which an iceberg can be found and if it is robust against the range of sizes that icebergs have. To this end a sensitivity analysis is done. This sensitivity analysis contains three tests. First we train the YOLOv5 algorithm using a training set of SAR images containing tabular icebergs around Antarctica, as identified and tracked by the USNIC. We train this model three times using different input image resolutions, in order to identify how sensitive the algorithm is to scale variations in the iceberg features and what feature scale is needed in order to distinguish icebergs from other objects in the ocean. We then test the sensitivity of the best performing model with respect to different atmospheric and oceanic conditions and consequent contrast variations, by detecting snipped out icebergs pasted on a variety of backgrounds (open ocean, rough weather, sea-ice). At last we also test the sensitivity of the model to variations in iceberg size by detecting inflated and deflated versions of the same iceberg pasted on one background. The result of this sensitivity analysis should then answer the following questions:

- How is the performance of YOLOv5 affected by changes in the spatial detail of an iceberg?
- How is the performance of YOLOv5 affected by changes in contrast between the iceberg and the surrounding area?
- How is the performance of YOLOv5 affected by changes in the size of an iceberg?

In relation to these questions we also investigate how the architecture and parameter settings of the YOLOv5 algorithm might lead to the obtained result and what the effect of the nature of the training data is on the results. This should give an indication of the possibility of using YOLOv5 for future iceberg detection and tracking studies.

The thesis is organized as follows: chapter 2 presents the satellite imagery used for training and testing the algorithm. It also presents the methods used for creating the augmented test data. Hereafter in chapter 3 an overview is given on the YOLOv5 algorithm and it is discussed how this algorithm is trained in order to be usable for the iceberg detection. Hereafter the testing setup is presented, which comprises of three sensitivity analyses. Chapter 4 shows the main results of our test setups. These are further discussed in chapter 5. In which their relevance is also viewed in the broader context of iceberg tracking. The subsequent possibility to use YOLOv5 for further iceberg detection and tracking studies is also treated here. Chapter 6 then gives the main conclusions to be drawn from this study.

2

Data

This chapter presents the data used in this thesis. Section 2.1 gives details on the type of satellite imagery that is used both for training and testing. Section 2.2 shows the preprocessing and download steps undertaken in order to make the data functional for the YOLOv5 algorithm and sensitivity analyses. section 2.3 shows how the data is split in a training and testing set and how the augmented test dataset is obtained.

2.1. Sentinel-1 Imagery

In this thesis imagery taken by ESA's Sentinel-1 satellites was used. The first of this dual satellite constellation (Sentinel-1A) is operational since April 2014 and in April 2016 also the second satellite of the pair (Sentinel-1B) was launched. The combination of these two satellites give a revisiting cycle of approximately 3 days over the polar regions and the imagery is thus very useful for high temporal resolution iceberg tacking[28]. The satellites provide C-band SAR imagery, centered at a frequency of 5.405 GHz with a wavelength of 5.6 cm[28]. We use Ground Range Detected (GRD) data, with a spatial resolution of 40 meter/pixel. The imagery comes in several polarizations, we however only used the HH polarization products, as this is the most readily available product[13]. Furthermore, we have only used imagery taken in extra wide swath mode, as this is the mode that is used over the polar ocean[7]. Some pre-processing steps are already undertaken before downloading, namely: border noise removal, thermal noise removal, radiometric calibration and terrain correction[6].

2.2. Preprocessing and downloading the training, validation and test data

In order to download imagery containing tabular icebergs, the weekly updates of iceberg coordinates and sizes around the Antarctic, provided by the USNIC iceberg database was used. The USNIC names all icebergs larger than 20 sqNM, or longer than 10 NM and stores their locations. The naming is given by a letter on the basis of the quadrant around the Antarctic circle from which the iceberg originated (A,B,C or D), and a number which indicates which iceberg it is, in the sequential order of detachment.

Only four of the weekly location updates of the USNIC were downloaded for each year that the Sentinel-1 mission was running, resulting in approximately 3000 locations of icebergs, containing 91 unique icebergs. From this dataset random samples were taken in order to provide a list of around 300 separate dates and locations. Figure 2.1 shows the locations of these icebergs.

Using the GEE python API, for each given date and coordinate one full sized Sentinel-1 image, if there was one available within three months of the given date, was downloaded overlapping the location of an iceberg. Before downloading, the images were filtered using a Gaussian kernel with a size of three pixels, in order to reduce the influence of speckle. Furthermore, because of size constraints the GeoTIFF images were instantly converted to JPEGs. Any double images, resulting from the fact that icebergs may lie close to each other, are manually deleted. The resulting dataset contained 247 unique images of approximately 14000 by 14000 pixels, were one image can thus contain several of the USNIC

tracked icebergs. Figure 2.2 shows an example of one of the downloaded training images. The Python notebook used for the preprocessing and downloading is found here.

Apart from imagery containing icebergs, we also downloaded imagery without any icebergs, on which we could paste the test icebergs. These were downloaded from two locations in the Bellingshausen Sea, where a seasonal sea-ice cover exists (two yellow dots in Figure 2.1) and few icebergs appear. 79 images were downloaded over the course of 2015 to 2022, with around one image per month, in order to provide a great variety of sea-states. For each of these images the mean sea-ice fraction over the image footprint at the time of acquisition was also computed, originating from the NOAA twice-daily updated sea-ice concentrations dataset (Available in the NOAA AVHRR Pathfinder SST dataset), in order to provide information on the sensitivity of the algorithm to sea-ice occurrence.

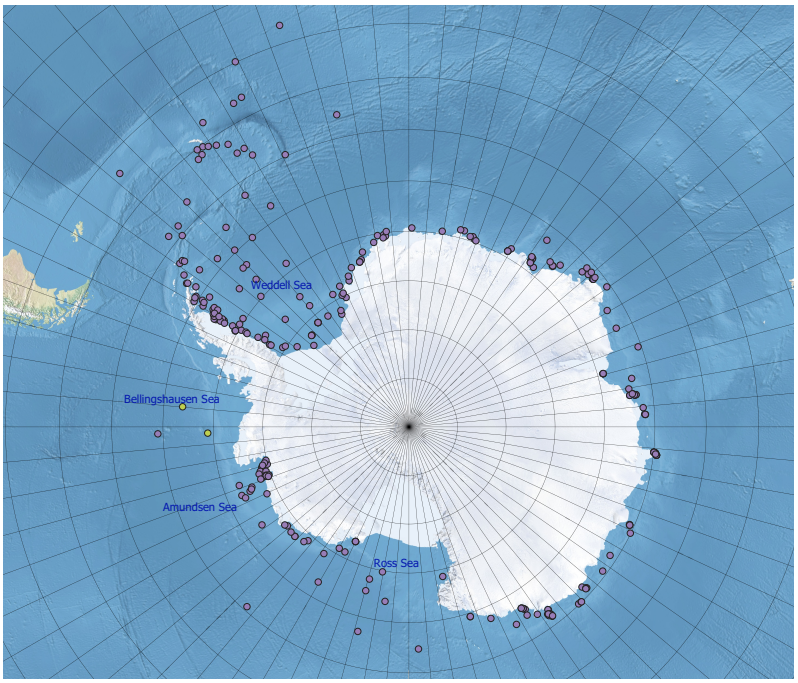


Figure 2.1: Locations of the USNIC tracked icebergs around Antarctica (purple dots) used to select the Sentinel-1 images for the training, validation and testing data and locations of the background images (yellow dots). Source: USNIC and Natural Earth

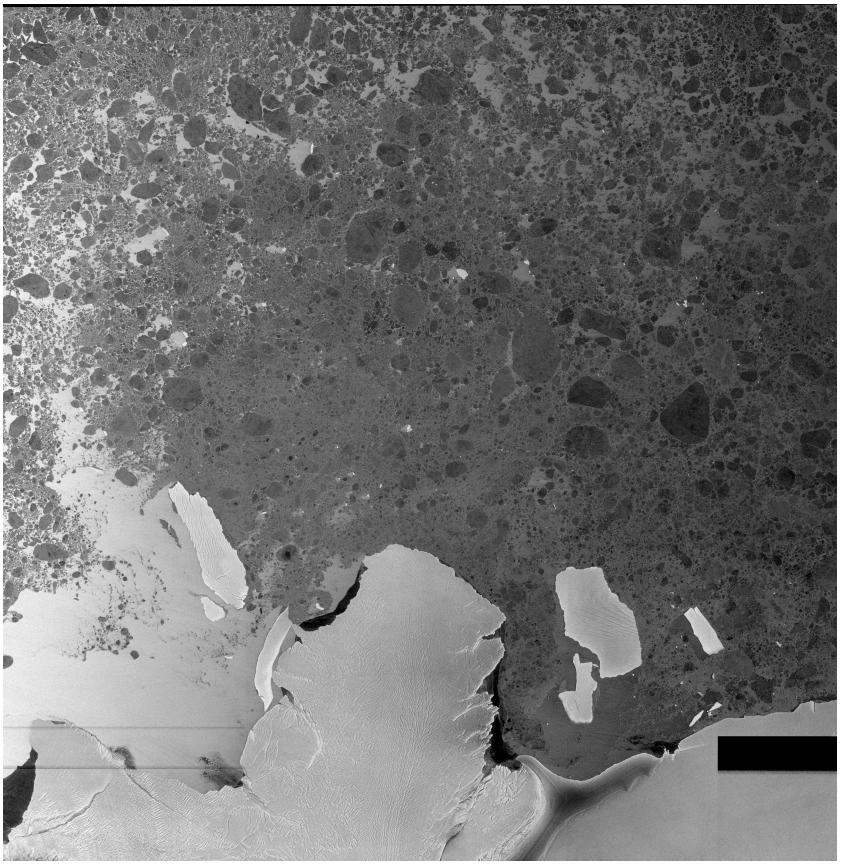


Figure 2.2: Example of one of the training images. Several large icebergs, tracked by the USNIC are visible in the frame, surrounded by ice floes, as well as some artefacts of either the scanning or processing of the scan.

2.3. Training, validation and test datasets

The 247 images with tabular icebergs were split up in a training, validation and test dataset. 65 icebergs in 146 images were used as training data (see Figure 2.3 for an overview of the appearances of these icebergs), 22 icebergs in 66 images were used as validation data (see Figure 2.4 for an overview of the appearances of these icebergs) and 4 icebergs (A68A, A68B, A68C and A74; see Figure 2.5) were used for the testing (sensitivity analysis on varying backgrounds and size). It was made sure that none of the icebergs present in the training data are present in the validation data or the test data and vice versa. It was also made sure that images were unique in order to prevent overfitting on one of the images. Furthermore, because in some cases a time interval occurred between the USNIC observation and the closest Sentinel-1 acquisition, some images contain none of the USNIC monitored icebergs, while others contain many.

All the USNIC monitored icebergs present in the imagery were annotated using bounding boxes, by means of the LabelMe application. The JSON files provided by this program were converted to YOLO compatible text files. These contain the class (iceberg), image coordinates of the center of the bounding box and the width and height of the bounding box for each of the icebergs. These values are given as a fraction of the full width and height of the image.

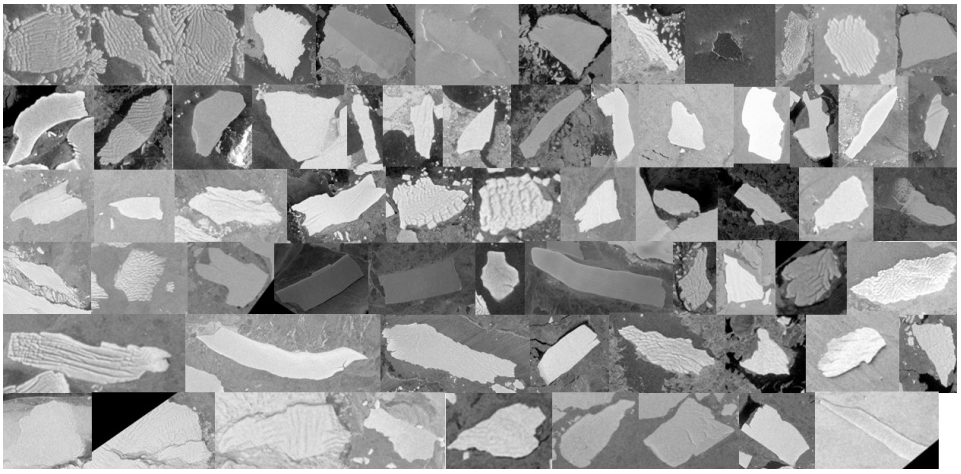


Figure 2.3: Overview of the icebergs found in the training data snipped out from full-sized Sentinel-1 images used for the training. The iceberg sizes are not according to scale.

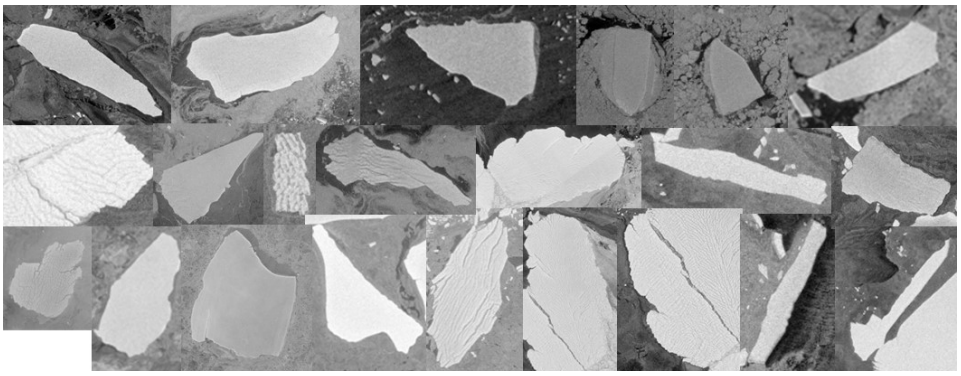


Figure 2.4: Overview of the icebergs found in the validation data snipped out from full-sized Sentinel-1 images used for the validation. The iceberg sizes are not according to scale.

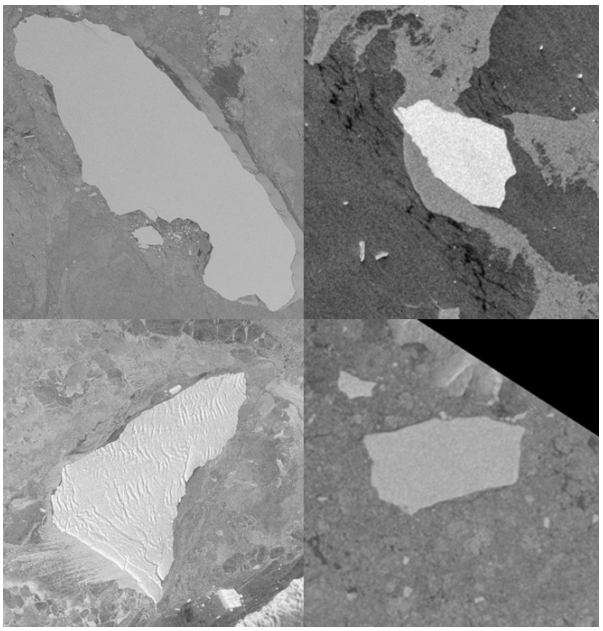


Figure 2.5: The test icebergs. Clockwise starting from the upper left: A68A, A68C, A68B and A74.

2.3.1. Test data

The test icebergs (A68A, A68B, A68C and A74; see Figure 2.5) were cut out from the original images. These icebergs show a variety of sizes, with lengths of the major and minor axis of these icebergs as shown in Table 2.1.

Iceberg	Major Axis (m)	Minor Axis (m)
A68A	158353	52565
A68C	12988	7341
A68B	15200	8800
A74	58165	34824

Table 2.1: Sizes of the test icebergs

For the sensitivity analysis with respect to changing atmospheric and oceanic conditions the four icebergs were pasted on all the 79 background images. As such certain physical appearance were mimicked, which should thus give an indication of the performance of the algorithm when applied in actual scenarios. Here some of the icebergs were partially cutoff from the image. As this poses another variable against which the algorithm could be sensitive, these images were not used for the analysis. The resulting dataset then contained 217 images (42 with A68A, 79 with A68B, 58 with A68C and 38 with A74). The bounding boxes surrounding these icebergs were also instantly computed in order to efficiently analyse the detection performance. This dataset is further mentioned as the background test dataset.

For the sensitivity analysis with respect to size we shrank and inflated every one of the four test icebergs using a range of factors as found in Table 2.2. The width and height of the original bounding box of the icebergs were multiplied with these factors in order to obtain the new size of the iceberg. The resulting icebergs were pasted upon one background, resulting in a dataset of 36 images. This dataset is further mentioned as the size test dataset.

The Python code used for creating the test data and the data itself can be obtained by contacting the author.

Iceberg	Shrinking factors
A68A	0.05, 0.15, 0.25, 0.35, 0.45, 0.55, 0.65, 0.75, 0.85, 0.95, 1.05, 1.15, 1.25
A68B	0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 4.0, 8.0
A68C	0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 4.0, 8.0
A74	0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 4.0

Table 2.2: Shrinking factors per iceberg, used for analysing the performance of the model on various iceberg sizes

3
Methods

This chapter presents the methods used, both for the training the algorithm itself as for the experiment setup and verification. Section 3.1 set outs the YOLOv5 algorithm that is used, and shows how the optimal weights of the model are obtained and chosen. Section 3.2 gives an overview on the experimental setups, used for the sensitivity analysis of the YOLOv5 algorithm.

3.1. YOLOv5

3.1.1. Overview and architecture

Object detection algorithms serve as purpose to detect, localize and classify objects in an image. Deep learning algorithms do this by use of neural networks. Such a network is generally built up as follows: an input image is fed into a series of CNN's which serve as feature extractors, this part of the network is called the backbone (the left box in Figure 3.1). Here various convolutional and pooling layers create high level feature maps, due to the activation of the kernel filters in the convolutional layers, these activate when spatial features of pretrained objects are present in the image. This backbone is in most cases built up as a common image classification network and should thus clearly define areas of interest where objects could be located. After this feature extraction, the exact localization and classification of the objects of interest takes place in the neck and head of the algorithm (center box in Figure 3.1). The localization is done, depending on the algorithm, by either a bounding box proposal surrounding the object, or a complete object segmentation.

In this research the Yolov5 algorithm is used[11], which is a family of object detection algorithms of various sizes. In this specific type of networks an input image first get segmented in boxes (x by x squares); each of these boxes serve as an anchor for a possible bounding box. In the backbone the network then extracts features, in three different levels (three arrows from Backbone to PANet in Figure 3.1), resulting in the identification of features with three sizes and consequent receptive fields. The receptive field indicates the spatial extent of which the feature extractor can hold information. A larger receptive field thus holds information on larger characteristics of the image. The three layers of feature extractors in the YOLOv5 models, indicate that the algorithm should, to a certain extent, be scale invariant. Furthermore, this backbone contains a Spatial Pyramid Pooling (SPP) layer. Because of this SPP layer the input image is not fixed to one specific size, which most other CNN's are, as the SPP layer can aggregate a range of tensor sizes to one fixed smaller size.

The actual proposal of the bounding boxes of an object occurs in the neck of the algorithm, where these objects are also instantly classified in the same part of the network. The candidate bounding boxes with the highest probabilities get selected as the final bounding box predictions using a non-max suppression layer. As such it is made sure that there are no pairs of bounding boxes, that define the same object. In the end, for each of the pretrained objects present in the image, the net gives an output consisting of a confidence score, classification, center location and bounding box extent (right box in Figure 3.1).

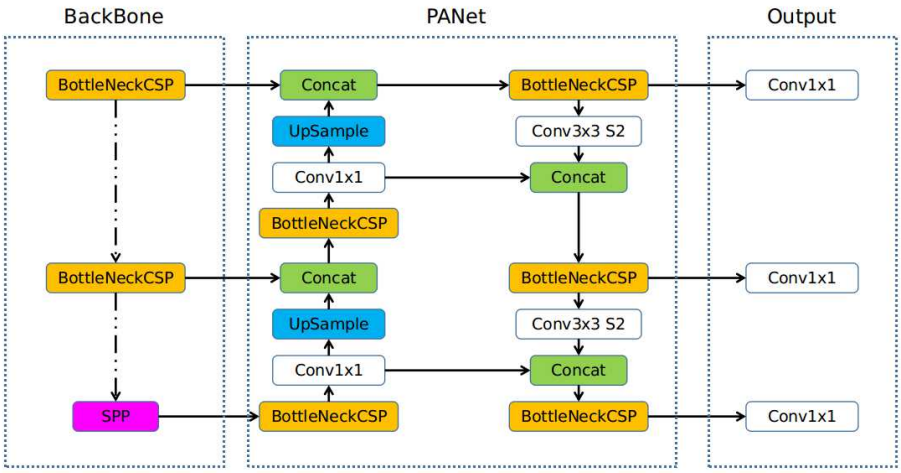


Figure 3.1: YOLOv5 Architecture overview. Source.

3.1.2. Training and optimizing the algorithm

Object detection algorithms that consist of neural networks need to be trained, in order to be able to detect objects of a class of interest in an image. In other words, the weights of the convolutional filters and fully connected layers inside the backbone, neck and head of the algorithm need to be optimized in order to be receptive of the spatial patterns that exist in the objects of interest. This is done by training the algorithm on images where the location and extent of the objects is known and labelled, i.e. the training data. One gives a batch of these images as an input to the algorithm, the algorithm then tunes the weights in its functions using gradient descent, such that the outputs of the algorithm have a larger probability to be equal to the true outputs of the batch. In general this process is repeated over several iterations of the whole input dataset (called epochs). Then after every epoch the partially optimized algorithm is tested against a labelled validation dataset, which is repeated until the optimal performance is obtained.

This training can be done either from scratch, i.e. training all the weights from zero, or using transfer learning, i.e. weights are already pretrained on other datasets/objects, which serve as a starting point for the training epochs. In this sense one can also transfer all the weights from another training round and choose to freeze some of them and thus only train a part of the network. The choice for either of these methods depend on the (dis)similarity between the objects of interest and the pretrained objects. The YOLOv5 models are pretrained on the COCO (Common Objects in Context) dataset, which consist of a large variety of common objects[5], not including icebergs. We did however choose to use this as a starting point for our training as the weights will likely also generate useful features for iceberg identification. In addition, training all these weights from scratch takes significantly longer, which was not deemed worth it given the preliminary nature of this research.

Model optimization

The optimal epoch of a model after a training run is measured using the mean average precision (mAP). The mAP is a frequently used performance metric in object detection. It is chosen because one wants to find the right balance between precision and recall (i.e. obtain the right balance between false positives and false negatives in the detection). The mAP is a measure for the area under the precision-recall curve. This curve is obtained by computing and plotting the precision and recall for each of the object classes for a range of confidence thresholds, for example the range: [0.1, 0.2, .., 0.9]. The precision and recall are computed as follows:

$$recall = \frac{TP}{TP + FN} \tag{3.1}$$

$$precision = \frac{TP}{TP + FP} \tag{3.2}$$

Here TP is the total amount of true positives in the dataset, FN the total amount of false negatives (thus objects that are not detected) and FP is the amount of false positives (objects that are detected as

belonging to a class, that do not belong to that class). A high recall thus indicates that most of the objects present in the image are correctly identified, while a high precision indicates that, from the detections, most are indeed correctly indicating an object that was present in the image. A prediction is TP, FN, or FP depending on the Intersection over Union (IoU) of the bounding boxes. The IoU is computed by dividing the intersected area of the detected and correct bounding boxes over the total unionized area of the two boxes, as shown in Figure 3.2. In order for a prediction to be labelled as correct the IoU should exceed a certain threshold. The value set for this threshold depends on the problem, as it depends on how accurate the localization of the object has to be.

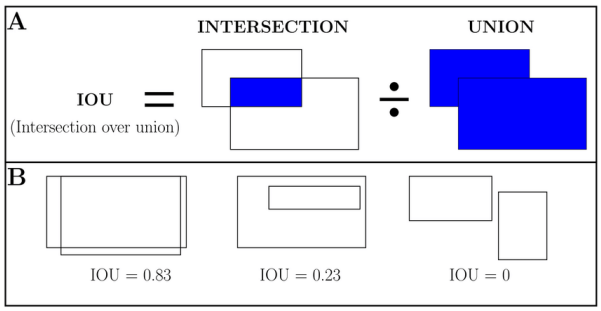


Figure 3.2: Example of the computation of the IoU. Source: Salau & Krieter, 2020 [21]

Using the predictions as obtained with a given IoU threshold, one then computes, for each of the object classes of interest, the AP over this whole curve through the following equation:

$$AP = \sum_{k=0}^{n} (recall(k) - recall(k+1)) * precision(k) \tag{3.3}$$

Here k is the index in the list of confidence thresholds, relating to a specific recall value. To obtain the mAP one takes the average of this AP over all the classes. However, as we deal with a binary classification (i.e. iceberg or no iceberg) there is only one class and the mAP thus is equal to the AP. The YOLO model, in particular assessed by means of the mAP[0.5:0.95] which means that the precision-recall curve and subsequent mAP is computed over a range of IoU thresholds from 0.5 to 0.95, with steps of 0.05. The average of all these mAP's is the mAP[0.5:0.95]. The optimal weights for the model are then chosen as the weights from the epoch where the mAP[0.5:0.95] is the largest.

There exist several versions of YOLOv5, all with differing sizes and thus various amounts of trainable parameters. For this research the second largest model was chosen (YOLOv5l), which consists of 47.8 million trainable parameters. A larger model, means the algorithm takes longer to train and detect, but the performance is in general also better. As the detection of icebergs does not need to be achieved in real time, a large model was chosen. There exist even larger models, these however were so large that using our image size, the hardware was not sufficient.

Parameter settings

The choice for the non trainable (hyper)parameter settings of the algorithm is something that can also be optimized. A large amount of parameters exist, which can be found here. We did however not optimize all the parameters in this study as this is a very time-intensive process and should not give significantly different results. Even without full optimization we should already be able to provide an indication of the possibilities that the algorithm could bring, when it is optimized completely.

3.2. Sensitivity analysis

3.2.1. Choice for confidence threshold in the sensitivity analysis

In order to do any testing with a trained model, one has to set a certain confidence threshold above which a prediction is given as output and beneath which the model discards a prediction. With low confidence scores there will thus likely be fewer false positives, and vice versa.

The choice for this threshold was based on the f1-score, which is the harmonic mean between the precision and the recall and a large f1 score should thus give a right balance between false negatives and false positives. It is computed as follows:

$$f1 = 2\frac{precision \cdot recall}{precision + recall} \tag{3.4}$$

For every confidence threshold this f1-core was computed on the basis of the validation data. The lowest confidence score at which this f1-score became stable with increasing confidence score was chosen as the optimal confidence score that was used as a threshold for the sensitivity analysis.

3.2.2. Test setup for the analysis of the sensitivity of the model to input image resolution

In order to gain insight in the sensitivity of the model to variations in spatial detail of the iceberg in the input images we trained the algorithm three times with three different settings for the input image size. This parameter determines how large the image is over which the algorithm trains or activates its feature extractors. A larger image input means that each pixel describes a smaller area, thus with larger image inputs the algorithm can be trained on smaller details of the icebergs. Three different image input sizes are tested: 640 x 640, 1600 x 1600 and 3200 x 3200. For our input images this relates to a pixel size of on average 750, 300 and 150 m/pixel, respectively. Even larger image resolutions were not possible due to computational constraints. The trained models are further mentioned as the 640, 1600 and 3200 models.

We compare the performance of these models both by using the validation data as well as by using the background test dataset (see subsection 2.3.1). By comparing both performances on the validation and test set it can be seen if a model possibly overfits on the training data. We compute the recall, precision and mAP for the validation set and the recall precision and f1-score for the background test dataset. After the comparison we choose the best performing model to be used for the subsequent sensitivity analyses on the basis of the largest f1-score.

3.2.3. Test setup for the analysis of the sensitivity of the model to different atmospheric and oceanic conditions

The second test setup was made in order to analyse the sensitivity to different atmospheric and oceanic conditions. We tested the optimal model found in the previous sensitivity analysis on the background dataset. We computed the recall per unique iceberg present in this dataset(A68A, A68B, A68C and A74 all separate), in order to obtain the best performing iceberg, which we investigated in the next part of this setup. As such the subsequent analysis of the reasons behind any false negative or false positive, relating to the surroundings, was less affected by the nature of the iceberg itself.

Using the data of the best performing iceberg we then investigated what the reasons of any false positives or false negatives were. This was done by first analysing the relation between recall, precision and the sea-ice fraction in the background image. For every background the mean sea-ice fraction was known. We then computed the recall and precision for any tenth percentile of this sea-ice fraction, in order to see if there is a strong correlation between sea-ice fraction and performance, for example all the false negatives could be occurring in images where a low amount of sea-ice exists.

Secondly, we also investigated if there was any apparent relation between performance and seasonality. Therefore we sorted the test dataset on the basis of the month in which the background image was taken. For each month we then computed the recall and precision.

Apart from this larger scale analysis we also manually investigated the imagery were false negatives and false positives occur. Here we searched for any apparent contrast or brightness situations in these images, either due to ice floes, sea-ice, land, rough sea conditions or SAR artefacts, by obtaining cross sections of the greyscale values over the icebergs and surroundings. These situations could be either very low contrast, inverted contrast between iceberg and ocean (the iceberg appears darker than the

ocean) or relatively bright icebergs with respect to the surroundings. This gave an indication as to whether the algorithm was robuster to changes in contrast than general thresholding methods.

3.2.4. Test setup for the analysis of the sensitivity to variations in iceberg size

The third test setup was made in order to investigate the sensitivity of the algorithm to different sizes of icebergs. This was done in two ways. We investigated the performance of the model with respect to the four test icebergs separately, as these already show a great range of sizes (see Table 2.1). We therefore computed the recall for every unique iceberg present in the background dataset as described in the previous section (subsection 3.2.3).

We also predicted the icebergs in the size test dataset (see subsection 2.3.1) using the optimal model. In order to assess the relation between performance and size, we plotted the major and minor axis of these inflated and deflated icebergs in a scatter plot, where the x and y axis are respectively the major and minor axis of the iceberg. Here it was also shown if the prediction of the iceberg is FN or TP. Furthermore, in order to assess if this performance was related to the architecture of the YOLOv5 algorithm or if it was the result of the specific training data, we also computed and plotted the lengths of the major and minor axes of all the training and validation icebergs. As such, if a cluster of training and validation sizes is apparent, and correct detections of the test set are only occurring when the major and minor axis of the test iceberg fall in this cluster, this might indicate that the algorithm was overfitted on only a small range of sizes. This setup should thus indicate if our algorithm is robust to multiple scales of icebergs, and if not, were the bottleneck might lie: in the training data or in the architecture of the algorithm.

4

Results

This chapter describes the results as obtained with the methods discussed in the last chapter. Section 4.1 describes the results of the model optimization. Section 4.2 shows results of the test with regards to the sensitivity to different input image sizes, as well as the choice for the best of the trained model models. Section 4.3 then shows the results of the performance and sensitivity test with respect to changing atmospheric and oceanic conditions. Section 4.4 shows the results of the performance and sensitivity test with respect to changes in size of the icebergs.

4.1. Model optimization

The training of the algorithm was done three times, with the different input image resolutions. We trained and tested the models on the GPU of the vrlab of the Delft University of Technology. The 640 model was trained for 150 epochs; the 1600 and 3200 model for 100. which took respectively 1.4, 4.9 and 17.9 hours. A larger amount of epochs was not tested, especially because the 3200 model already took very long to train. All the weights were trained except for the head of the algorithm. We thus froze the 10 lower layers of the neural net. Freezing the whole net except for the last layer, gave insufficient results, as the precision and recall were around 0.3. Training the whole net including the head took longer and did not give significantly better results (which was also noted by the author of the algorithm when training on the PASCAL VOC dataset, see here). All the default (hyper)parameter settings were used, which can be found here. One parameter that was altered however was the batch size, this parameter is generally set as high as possible[10]. Here a batch size of two was the largest batch size that could be used for all input image sizes due to memory limits of the hardware.

Figure 4.1 shows the mAP[0.5:0.95] curves of the three models. For all models the mAP increases fast over the first 50 epochs and afterwards it starts to stabilize. Even at the last epoch there still seems to be some increase in performance for all three; the mAP is thus not yet fully stable after the final epoch, although the performance does not seem to be increasing by a significant amount. The optimal models are chosen as the model after epoch 146, 96 and 96 for the 640, 1600 and 3200 model respectively, as these give the largest mAP[0.5:0.95] score.

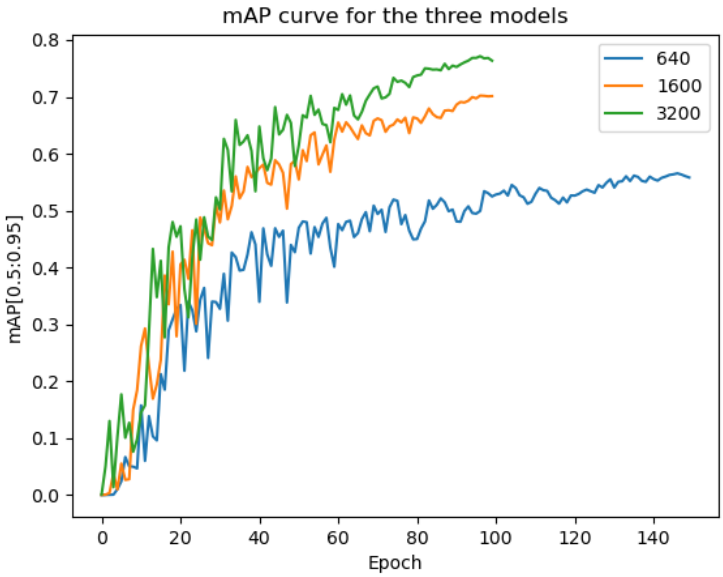


Figure 4.1: mAP curve of the three different models. The Y-axis shows the mAP score computed on the validation data after every epoch (X-axis). A sharp increase is visible in the first 50 epochs, after which the increase in performance becomes lower up until the final epochs. In the last few epochs there still is some increase in mAP visible.

The f1-confidence threshold plots for the optimal model weights are given in Figure 4.2. After a confidence score of approximately 0.3, the f1-score stabilises or decreases with increasing confidence score threshold, hence a confidence score threshold of 0.3 was chosen for the next steps in the analysis. On top of this we consider a detection as correct if the IoU between the bounding box prediction and the correct bounding box is larger than 0.5.

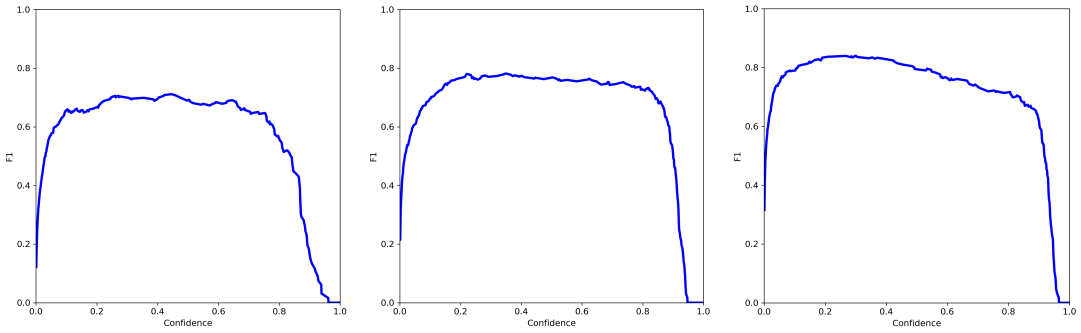


Figure 4.2: F1-confidence curves of the three models. From left to right: 640, 1600 and 3200 model. Y-axis shows the f1-score and x-axis shows the range of confidence scores. For all three models the curve increases sharply from 0 to 0.2 and then stabilizes until an optimum around 0.3, before decreasing slightly until a confidence score of around 0.8 and then dropping of sharply

4.2. Results of the sensitivity analysis with respect to input image resolution

The performances of the three optimized models on the validation set are found in Table 4.1. The precision, recall and mAP are all the largest for the 3200 model. The recall of the 1600 model is slightly lower than of the 3200 model, while its precision is slightly lower than of the 640 model. The best performing model on the basis of the validation data thus is the 3200 model. The performances of the three models, with the given IoU and confidence thresholds on the background test dataset are found in Table 4.2. The precision, recall and f1-score are all the largest for the 3200 model. The precision of the 1600 model lies right in between that of the 640 model and 3200 model, while the recall is comparable to that of the 640 model. It is thus apparent that with an increase in image input resolution the performance

gets significantly better, both in terms of a false positives and false negatives. It is furthermore notable that the 1600 model does perform better than the 640 model in terms of precision, but not in terms of recall. This thus shows an opposite relation than for the validation data. The recall only significantly increases when the input image size is 3200 pixels. As the f1-score is the largest for the 3200 model we chose to use this model for further analysis.

Model	Best Epoch	Precision	Recall	mAP[0.5:0.95]
640	146	0.759	0.669	0.566
1600	96	0.731	0.857	0.702
3200	96	0.826	0.866	0.772

Table 4.1: Performance of the models on the validation dataset

Model	Precision	Recall	F1-score
640	0.55	0.45	0.49
1600	0.76	0.49	0.60
3200	0.90	0.71	0.79

Table 4.2: Performance of the models on the background test dataset

4.3. Results of the sensitivity analysis with respect to atmospheric and oceanic conditions

The 3200 model was tested on the background test dataset. The recall per iceberg subset of this dataset is shown in Figure 4.3. The recall for iceberg A68A is zero and this iceberg is thus not detected at all. The recall for the A68B iceberg is the largest, slightly better than the recall of the A68C and A74 iceberg. The A68B iceberg is thus most easily detected. We therefore chose this iceberg for the subsequent sensitivity analysis of the model with respect to different atmospheric and oceanic conditions.

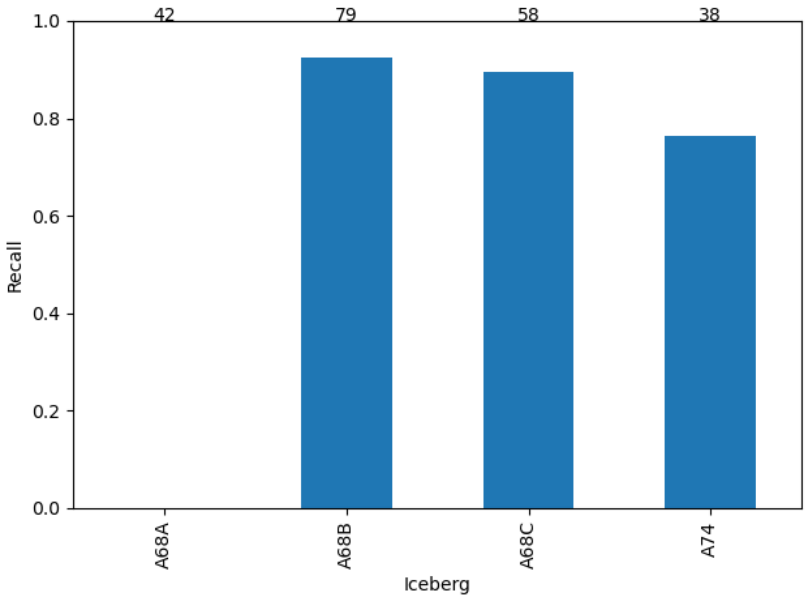


Figure 4.3: Recall of the 3200 model for each of the four test icebergs with different backgrounds. The number on top of the bar indicates the number of images present per iceberg.

For each background on which we pasted the A68B iceberg, we computed the mean sea-ice fraction. Figure 4.4 shows the recall and precision achieved on these images computed per tenth percentile sea-ice fraction interval. There does not seem to be a significant relationship between sea-ice fraction

and precision or recall. All false positives and false negatives seem to be irregularly distributed over the sea-ice fraction range. It should however be noted that the distribution of imagery over the interval is not regular and therefore any trend that on a large scale could be occurring might be obscured by seemingly random variations in the small dataset.

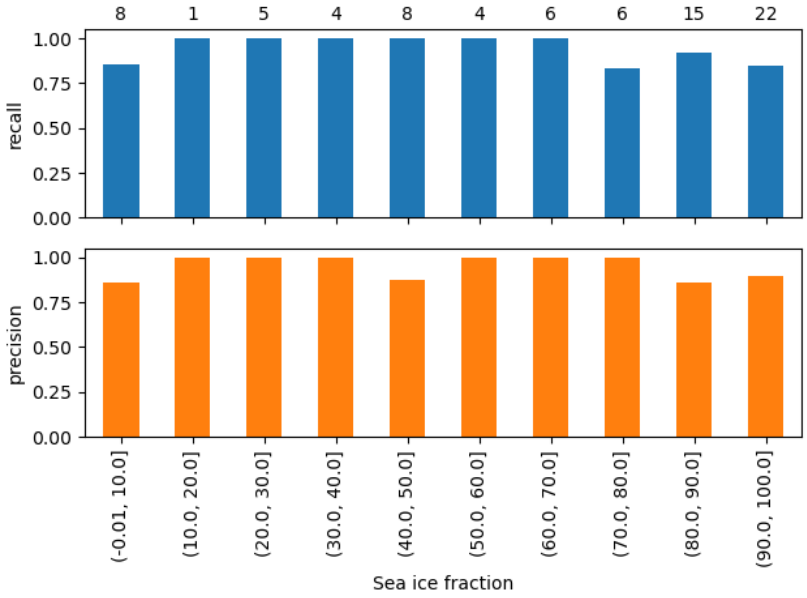


Figure 4.4: Recall and precision of the 3200 model for imagery containing the A68B iceberg on top of varying backgrounds, of which the mean sea-ice fraction is known as derived from the NOAA sea-ice product. The number on top of the bars indicates the number of images present per binned mean sea-ice fraction.

The monthly statistics of detection performance are found in Figure 4.5. Here the precision and recall is lowest in the austral winter (July to October) and in other months the recall and precision are (close to) one. The model thus performs best in summer and worst in winter.

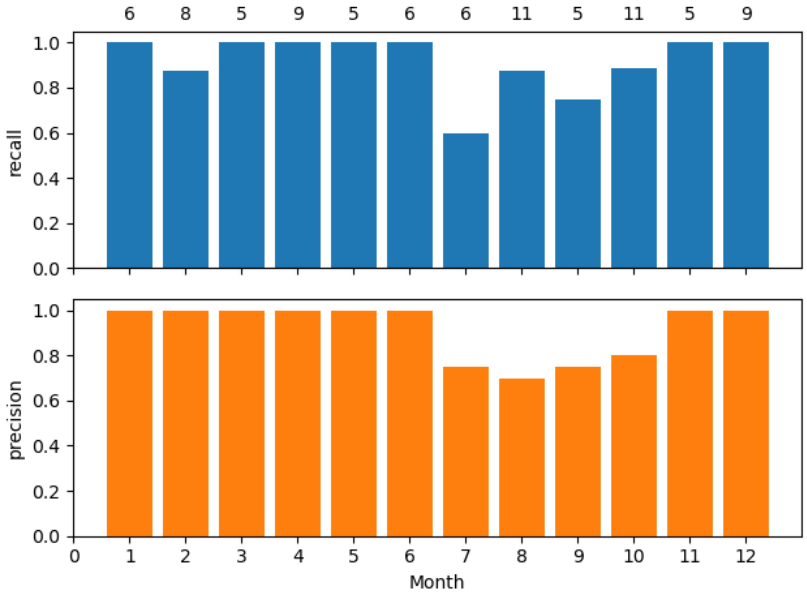


Figure 4.5: Recall and precision of the 3200 model for imagery containing the A68B iceberg on top of varying backgrounds, per month. The number on top of the bars indicates the number of images present per month. The precision is lowest in the months of July until October (lower graph). The same goes for the recall, but here one outlying recall value appears in February.

The imagery of six of the seven of these false positive predictions occurring in the austral winter are shown in Figure 4.6. In this figure prediction A, B and D show the same circumstances, namely very rough atmospheric conditions. These conditions induce the appearance of objects that appear very bright and homogeneous – thus a similar appearance as icebergs – which are likely the result of snow storms. False predictions E and F are the result of frozen ice-floes that show a similar geometry and brightness as icebergs. Prediction C actually is a correct prediction of an iceberg, however because this iceberg was part of the background on which the A68B iceberg was pasted, it was not registered and therefore it came out as false positive.

The model thus, in terms of false detections, indeed seems to be sensitive to the atmospheric and sea-state conditions during winter, but it should be noted that there are also many cases of ice floes and rough weather conditions were no false detections took place, as is apparent from the still large precisions between 0.7 and 0.8 in these winter months.

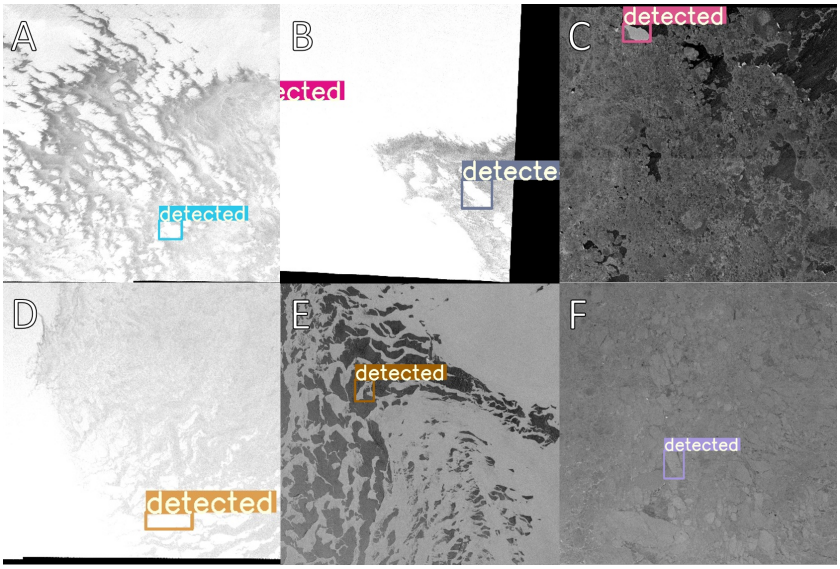


Figure 4.6: Six of the false positive detections. A, B and D show very rough atmospheric conditions resulting in a bright image. C actually shows a true positive as this is indeed an iceberg, but one that was present in the background image. E and F show conditions with many frozen ice floes that contain shapes reminiscent of icebergs.

Figure 4.7 show the images in which the model did not detect the A68B iceberg. The false negatives occur in conditions were the relatively dark A68B iceberg is amid brighter spatially homogeneous surroundings, either due to sea-ice (covered by snow) or atmospheric conditions (snow-storm). It thus seems that a situation were the iceberg has a lower brightness than its surroundings is problematic for the model, and a clearly positive contrast between iceberg and surrounding might be needed to detect an iceberg.

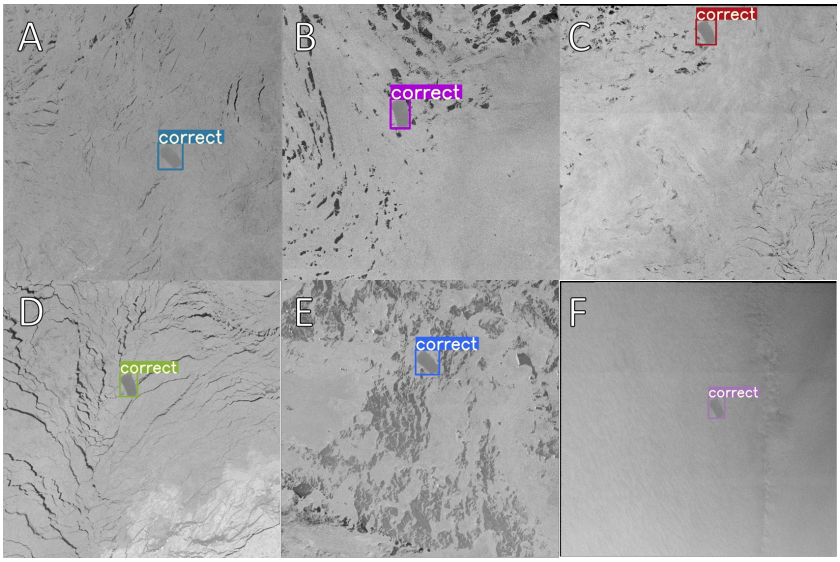
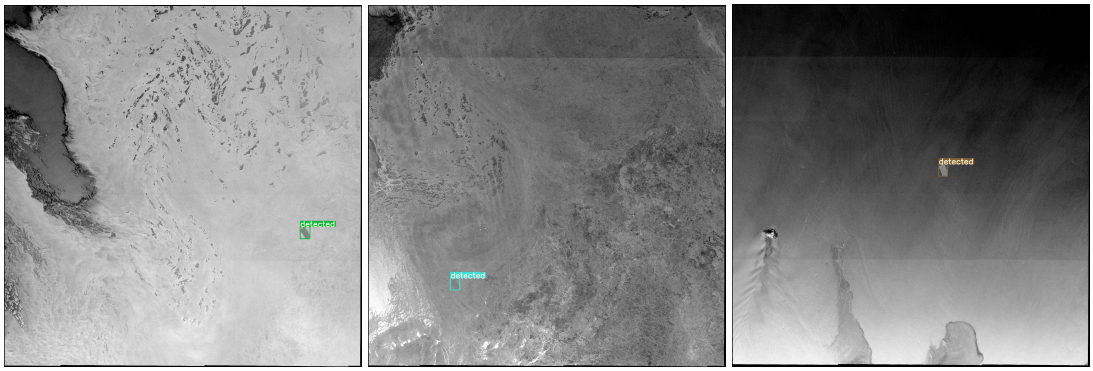


Figure 4.7: Six (all) of the false negative detections. In all images the iceberg appears darker than the bright surroundings.

Figure 4.8a however shows a correct detection of an iceberg that appears darker than its surrounding. Furthermore Figure 4.8b shows a correct detection of an iceberg in a very low contrast situation. Here the iceberg is hard to distinguish manually, but the model does not show any trouble with the detection. Figure 4.8c also shows a situation where the iceberg appears brighter than the surroundings and is correctly detected. Consequently it seems that the YOLOv5 model can, in contrast to what seems apparent from Figure 4.7, distinguish icebergs independent of a fixed positive or negative contrast situation.

We plotted the greyscale values corresponding to the measured backscatter along the minor axis of the A68B iceberg and its surroundings in the images from Figure 4.8 and Figure 4.7C. The results are shown in Figure 4.9. The greyscale values along the minor axis of the only false detection (from Figure 4.7C) lies in between the profiles of the other correct detections, as such it seems that the model can identify icebergs that show both a negative and positive contrast with its surroundings. Furthermore, in very low contrast situations a correct detection is also possible (blue line in Figure 4.9. It should be noted that the variance in brightness of the iceberg is significantly smaller than the variance in brightness of the surroundings. This could thus be one of the factors that the YOLO model uses for the detection. The one false negative (red line in Figure 4.9) however shows approximately the same difference in variance and the cause of the false negative hence is likely not due to any dissimilarity with respect to this factor. The factor that might be making the distinction is the sharp drop in brightness around the 400th pixel (a darker spot in Figure 4.7C at the lower left of the iceberg). This however is a very local spot and does not occur at all the other false negatives, a clear relation can therefore not be made. It thus seems that the appearance of false negatives does show relation with the background situation, as all the false negatives occur in situations where they are surrounded by brighter ocean/ice, but it is explicitly clear that this relationship is caused by the appearance of a negative contrast.



(a) Correct detection of dark A68B iceberg amid a very bright background

(b) Correct detection of dark A68B iceberg amid an equally dark background. Even with the naked eye the iceberg is hard to distinguish.

(c) Correct detection of the A68B iceberg appearing bright amid a very dark background.

Figure 4.8: Three significantly different conditions in which the iceberg is detected correctly.

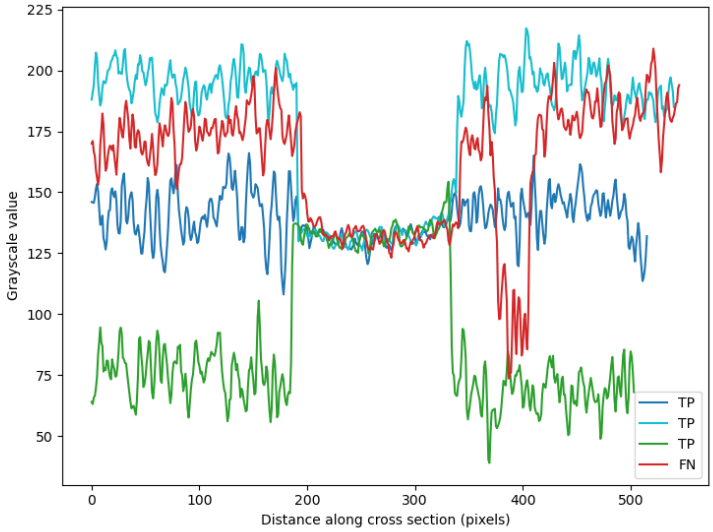


Figure 4.9: Greyscale values along minor axis cross sections of the A68B iceberg in four different scenarios: the images from Figure 4.8 and Figure 4.7C. The red line indicates the only non-detected iceberg of the four. The iceberg is contained in the distance 200 to approximately 330. The FN prediction shows a contrast pattern where the surroundings have a larger backscatter than the iceberg itself. The same goes for one of the TP detections (light blue line). The other TP detections show a low contrast situation (dark blue line) and a positive contrast situation (green line, backscatter of the iceberg is larger than of the surroundings). The FN detection thus shows a contrast pattern that lies in between the contrast patterns of the TP detection and thus does not show a clear outlying contrast relation compared to the other TP detections.

4.4. Results of the sensitivity analysis with respect to size

Figure 4.3 shows the performance of the model on each of the four test icebergs with different sizes. The test resulted in recalls of 0, 0.92, 0.90 and 0.76 for iceberg A68A, A68B, A68C and A74 respectively. If we compare this performance to the sizes of the iceberg, as found in Table 2.1, it is apparent that the largest iceberg (A68A) achieves the lowest recall – it is never detected. Furthermore the second largest iceberg (A74) shows a significantly greater recall, but still performs worse than the two smaller and approximately equally sized icebergs (A68B and C). Size thus seems to influence the performance of the YOLO model significantly.

Figure 4.10 shows the major and minor axis of all training and validation icebergs plotted against each other, together with the original sizes of the test icebergs and the derived sizes of the test icebergs

after multiplying their size with the shrinking factors as found in Table 2.2. TP detections are marked green and FN's red.

All correct detections fall inside one particular area (green ellipse in Figure 4.10) which is the area where the largest cluster of training and validation icebergs occur. The test icebergs with major and minor axes smaller than the smallest training icebergs (lower left corner of plot, major axis < 5 km) are not detected, even as the icebergs with major and minor axes significantly larger than the icebergs in this cluster (major axis > 75km). Although it should be noted that some larger icebergs (e.g. iceberg A74) are correctly detected while their sizes do not fall inside the dense cluster. The results from the shrinking and inflation test using the size test dataset thus show the same sensitivity to size as was apparent from investigating the four test icebergs separately.

Furthermore it is notable that the A68A iceberg (which was never detected in the original dataset) has a major and minor axis far greater than nearly all of the training icebergs, which will likely be the reason that it was not detected. As can be seen (following the grey line through A68A) the smaller versions of this iceberg are detected. For the initially smaller icebergs (A68B and C), inflating them causes the model to not detected them anymore, thus implying that it is not the shape that makes icebergs detectable, but indeed the size.

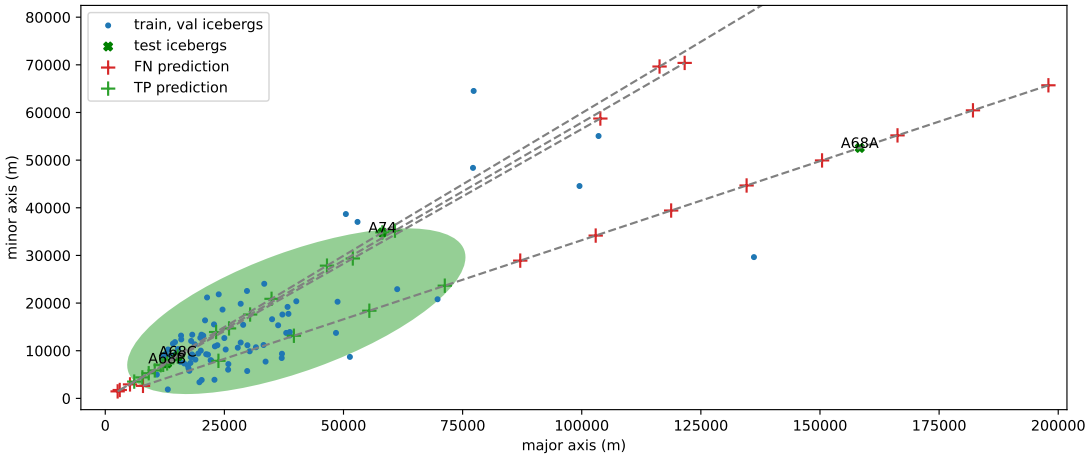


Figure 4.10: Plot of the major axis length against minor axis length, for all the training, validation and test icebergs. Green ellipse indicates the range of sizes where correct detections occur. The grey lines are the trajectories in size of the four test icebergs, as a result of the range of shrinking factors applied. The green crosses and names, correspond to the original sizes of the test icebergs.

5 Discussion

In this chapter the results are discussed. In section 5.1 the performance differences between the three trained models and related sensitivity to input image resolution are discussed. In section 5.2 the sensitivity of the model to the varying background states is explained, while section 5.3 explains the variations in performance with varying iceberg sizes. Section 5.4 shows recommendation for future research and future applications of the YOLOv5 model. Here we also put the detection results in the context of iceberg tracking.

5.1. Sensitivity of the algorithm to variations in input image resolution

We trained three different models, with different image input sizes (640x640, 1600x1600 and 3200x3200). It was found that the largest image input size held the best performance, both on the validation and the testing data. The recall was however better on the validation data than on the test data, while the precision showed an opposite relation. The fact that the recall was better on the validation data suggests that the validation and test data show different characteristics and the model is thus slightly overfitted on the validation icebergs. The fact that the precision is higher on the test data also suggest different characteristics of the area surrounding the icebergs. It is likely that the lower precision for the validation data is either caused by the appearance of smaller icebergs around the USNIC tracked tabular icebergs, which are not labelled, or the presence of islands or glaciers/ice shelves that might appear as icebergs. These are both not appearing in the backgrounds of the testing data. These slight differences in optimum recall and precision do however not seem to be strongly related to the sensitivity of the model to changes in input image size, as they occur with every input image size.

What is interesting to notice in relation to the input image size is the fact that larger images (and consequent smaller pixel size) result in better performance. This is likely the result of the fact that the algorithm will train on smaller features when input pixel size is smaller, because the receptive fields of the feature extractors will contain the same amount of pixels, independent of the image input size. These smaller scale features thus seem to be more capable at characterising icebergs and distinguishing them from other objects like islands or weather phenomena. This seems very obvious when one imagines a small island and an iceberg in a blurred image. The sharp corners often apparent in icebergs will then seem more rounded, which also most often characterises islands. The results thus indicate that the algorithm is indeed very sensitive to the image size parameter and when using YOLOv5 for iceberg detection, it is seems likely that one needs to use the largest image size that the hardware is capable of, in order to obtain the optimal performance. Although it should be noted that using even larger imagery (and smaller pixel size) could also cause overfitting to very small scale features apparent in the training data, this problem could however be overcome if a larger enough training dataset is used.

5.2. Sensitivity of the model to variations in atmospheric or oceanic conditions

The result suggest that the performance of the model is slightly sensitive to the presence winter conditions. It is however quite robust against a range of other atmospheric and oceanic conditions and

even in most winter images, the detections are correct. We did however find that the precision and recall was lower in the austral winter than in the other months.

Upon inspection of the false positives of these winter months it was found that these were indeed related to the presence of winter phenomena: frozen ice floes and rough weather conditions (i.e. the presence of wind and snow). These conditions give rise to objects that might in shape and brightness appear as icebergs. This is a problem which Wechse and Dierking (2012) also noticed, they suggested that the best conditions for iceberg detection (using conventional methods) is in low wind speed and freezing conditions, but only if these freezing conditions do not result in pancake shaped ice floes[30].

Our YOLOv5 model thus does not completely mitigate these proposed problems, but exploring the false negative and true positive detections it can be seen that there are situations in which it seems that the conventional thresholding methods would not necessarily be able to detect the iceberg, but the YOLO model does. As the results showed there were low contrast, negative contrast and positive contrast situations in which icebergs were all detected. These situations could in reality relate to imagery containing either rough weather/ocean conditions, melting icebergs, snow covered sea-ice or SAR artefacts, all problematic situations from which conventional methods suffered[26, 30, 19, 17].

The YOLO model could thus potentially overcome these problems, especially if the model is trained on a larger and more diverse dataset, as the dataset used here is very small in comparison to most deep learning training datasets and we did not use any data augmentation (contrast, brightness, zoom changes). It might then be possible to overcome even the few false negatives that occurred in the test, as it was shown that the backscatter and spatial signatures of these false negatives do not differ too much compared to correct detections. A slight optimization of the feature extractors might thus be enough to be able to detect these.

Nevertheless, it should be noted that the test data used here was augmented data, which thus by definition differs from real imagery and results can therefore not be completely extrapolated to a real case. Figure 4.9 showed that the variance in brightness in the pasted iceberg was significantly lower than the surrounding variance. This is also apparent in real SAR imagery of icebergs, as it is one of the phenomena exploited for conventional detection methods, but it might be that because of the pasted nature of the iceberg, the contrast between background and iceberg is more obvious and thus easier for the model to be identified. If the latter is indeed true for our data is not a question that is investigated here, but it is something to be aware of, which could cause the actual detection in future testing of real scenario's to be of lower quality.

Furthermore, in the scenario's of rough weather situations, it might in reality be tougher to identify the icebergs. In the real case the falling snow transported by wind sits on top of the icebergs and thus could fully obscure the iceberg, while in our data we paste the iceberg on top. The latter means that the contrast between the iceberg and its surroundings could be as low as it is in reality, but the variance in backscatter at the location of the iceberg might still be lower than it would be if it was obscured by falling snow, thus resulting in a larger possibility of detection. Further testing using real imagery is thus needed to identify the complete sensitivity to these rough weather scenarios.

5.3. Sensitivity of the model to variations in iceberg size

Size is, at least in our model and test setup, the most defining factor for the performance of the YOLOv5 algorithm as it was found that very large icebergs (major axis > 75km) and very small icebergs (major axis < 5 km) were never correctly detected and the intermediate sized A74 iceberg also returned weaker performance than the two smaller A68B and C icebergs.

This shows that the spatial and spectral features that the model uses for the detection of icebergs do not appear in very small icebergs or very large icebergs. For small icebergs the reason behind this is very apparent: these icebergs have axis lengths around 5000 m and with a pixel size of 150 m, these will thus only incorporate at maximum around 30 pixels along one edge. This likely makes it hard to identify any distinctive spatial feature and thus detection does not occur.

Another reason for not detecting these smaller icebergs also explains the lack of detection of larger icebergs: the training set mostly consists of icebergs within a certain size distribution and is in that sense not very equally distributed, with only a small amount being either very large or very small. This means that the features, which the model identified during training, are optimised for this particular size of icebergs. These features do not seem to be scalable to larger (or very small) icebergs. To a human it seems that smaller and larger icebergs do show approximately the same shapes: the same abundance

of either rounded corners or sharp pointy corners, together with longer edges. These seem to be present on the same scale in both large and small icebergs, but larger icebergs just show a larger amount or more spread out distribution of these shapes. It therefore seems that the model is not incorporating these edges and corners as features by itself. It is more likely that the model is identifying a combination of several of these features, relating to possibly a full extent of a small iceberg. As such larger icebergs will not be detected, because their edges are just too far apart to activate one such feature map.

It seems likely that this problem of scalability can be solved by training the algorithm on a more evenly distributed dataset. It might then train to identify features that are both apparent in larger and smaller icebergs. It could however also be that these features are then not distinguishing icebergs well enough from other objects, thus lowering the performance. One way to solve this might be by defining two classes of icebergs, one with large icebergs and one with smaller icebergs. These are then defined by different features thus making the detection more scale invariant. A last problem could however be that the receptive field of the feature extractors is just too small to be able to define larger features in large icebergs, as such the model might still only identify smaller features that do not distinguish these larger icebergs well.

Iceberg size is thus an important factor influencing the performance of the detection, but the results do suggest that this is a factor which can be accounted for if the training data or algorithm itself is optimized.

5.4. Recommendations for future use of YOLOv5 for iceberg detection and tracking

The YOLOv5 model shows great potential for the detection of icebergs, as it seems quite capable of detecting icebergs in a variety of situations, relating to different surface and atmospheric conditions and it is independent of any thresholds set on the basis of manual inspection of the icebergs of interest.

Still, this research only serves as a first hand identification of the possibilities of the algorithm and several potential problems relating to detection and tracking of icebergs still have to be investigated or optimized, in order to say if this algorithm serves as a great alternative to conventional methods. First of all more test using different algorithm setups (as mentioned in section 5.3) have to be done in order to explore the potential scale invariance of the algorithm, which is essential for correct automated detection of the wide variety of iceberg sizes. If all sizes of icebergs can indeed be detected using this algorithm, it could give great possibilities for the estimation of the total ice discharge. Secondly it has to be explored if this algorithm can also identify icebergs that are partially cutoff from the image, which is a situations often encountered when tracking icebergs. Thirdly, the algorithm also has to be tested on real data and the amount of training data could also be expanded, both by using more images and data augmentation as this will likely increase the performance and reduce any overfitting. Apart from these tests, the algorithm settings itself also need to be further optimised, for example the amount of training epochs or even image size could be further increased, which could result in greater performances.

Aside from the results obtained here and the potential further testing of the model in relation to iceberg detection, the ultimate goal is using these deep neural nets for the tracking of icebergs. YOLOv5 in combination with DeepSORT could indeed potentially be used for this case as our results indicate that at least YOLOv5 seems quite insensitive to the wide range of spectral and spatial signatures of icebergs and their surroundings. The next step is testing whether DeepSORT can also match these detected icebergs, even with their frame-to-frame variations in spectral signature. Aside from the combination with DeepSORT, another possibility of tracking using YOLOv5 might be by detecting the icebergs frame-to-frame through YOLOv5 and then segmenting these detected bounding boxes using some segmentation and matching methods as f.e. done by Koo et. al (2021)[13]. This might enhance the performance they achieved, as one now need to segment only a small portion and it is already known that an iceberg will very likely be located there.

6

Conclusion

In this project the applicability of the YOLOv5 object detection algorithm on the detection of icebergs in SAR imagery was explored. To this end a sensitivity analysis on its performance under various input image resolutions, atmospheric and oceanic conditions surrounding the iceberg and iceberg sizes was performed. The results seem very promising. It was found that with a larger input image resolution (and related smaller pixel size) the model performed significantly better, making it clear that icebergs are well distinguishable from other objects in the polar oceans on the basis of small scale spatial features. It was also found that under various contrast scenarios, relating to different ocean and atmospheric states surrounding the icebergs, the icebergs could in most cases be detected. Consequently, the obtained model seemed more robust to differing contrasts between the iceberg and its surroundings than conventional thresholding methods used for iceberg detection. The performance of the model however seemed to be very sensitive to the size of the icebergs as it could only detect icebergs inside a particular size range. This was likely the result of the unequally distributed size range in the training data. A more equally distributed dataset might therefore give a significantly better performance on a wider range of sizes. The architecture of the algorithm might however pose some constraints on the extent to which this performance can be enhanced. Further testing on real-life imagery and scenarios is still needed in order to give a decisive comparison to current detection methods, but the results found here indicate that YOLOv5 and deep neural nets in general bring great possibilities for the detection of icebergs. Apart from that the next step is using the YOLOv5 model in combination with DeepSORT or other tracking methods, in order to test whether the matching of icebergs in consecutive SAR images can also be performed under the variations of backscatter and geometric iceberg signatures.

References

[1] Imran Ahmed and Gwanggil Jeon. "A real-time person tracking system based on SiamMask network for intelligent video surveillance". In: *Journal of Real-Time Image Processing* 18.5 (2021), pp. 1803–1814.

[2] Mauro M Barbat et al. "Automated iceberg tracking with a machine learning approach applied to SAR imagery: A Weddell sea case study". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 172 (2021), pp. 189–206.

[3] Mauro M Barbat et al. "Three years of near-coastal Antarctic iceberg distribution from a machine learning approach applied to SAR imagery". In: *Journal of Geophysical Research: Oceans* 124.9 (2019), pp. 6658–6672.

[4] Jeffrey S Budge and David G Long. "A comprehensive database for Antarctic iceberg tracking using scatterometer data". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11.2 (2018), pp. 434–442.

[5] COCO. *COCO Common Objects in Context*. 2022. URL: `https://cocodataset.org/#home` (visited on 03/27/2022).

[6] Google Earth Engine. *GEE Guides, Sentinel-1 Algorithms*. 2022. URL: `https://developers.google.com/earth-engine/guides/sentinel1` (visited on 04/21/2022).

[7] ESA. *Sentinel-1 User Guides, Acquisition Modes, Extra Wide Swath*. 2022. URL: `https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-1-sar/acquisition-modes/extra-wide-swath` (visited on 04/21/2022).

[8] NF Glasser and Ted A Scambos. "A structural glaciological analysis of the 2002 Larsen B ice-shelf collapse". In: *Journal of Glaciology* 54.184 (2008), pp. 3–16.

[9] Yang Jie et al. "Ship detection and tracking in inland waterways using improved YOLOv3 and Deep SORT". In: *Symmetry* 13.2 (2021), p. 308.

[10] Glenn Jocher. *Tips for Best Training Results*. 2022. URL: `https://github.com/ultralytics/yolov5/wiki/Tips-for-Best-Training-Results` (visited on 04/21/2022).

[11] Glenn Jocher. *YOLOv5 Github*. 2022. URL: `https://github.com/ultralytics/yolov5` (visited on 03/27/2022).

[12] Juha Karvonen et al. "Iceberg Detection in Dual-Polarized C-Band SAR Imagery by Segmentation and Nonparametric CFAR (SnP-CFAR)". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2021), pp. 1–12.

[13] YoungHyun Koo et al. "Semi-automated tracking of iceberg B43 using Sentinel-1 SAR images via Google Earth Engine". In: *The Cryosphere* 15.10 (2021), pp. 4727–4744.

[14] Christoph Lichey and Hartmut H Hellmer. "Modeling giant-iceberg drift under the influence of sea ice in the Weddell Sea, Antarctica". In: *Journal of Glaciology* 47.158 (2001), pp. 452–460.

[15] Hai Lin et al. "Free-drifting icebergs as sources of iron to the Weddell Sea". In: *Deep Sea Research Part II: Topical Studies in Oceanography* 58.11-12 (2011), pp. 1392–1406.

[16] David G Long, Jarom Ballantyn, and Cheryl Bertoia. "Is the number of Antarctic icebergs really increasing?" In: *Eos, Transactions American Geophysical Union* 83.42 (2002), pp. 469–474.

[17] Ludwin Lopez-Lopez et al. "On the detection and long-term path visualisation of a-68 iceberg". In: *Remote Sensing* 13.3 (2021), p. 460.

[18] Seelye Martin, Robert S Drucker, and Ronald Kwok. "The areas and ice production of the western and central Ross Sea polynyas, 1992–2002, and their relation to the B-15 and C-19 iceberg events of 2000 and 2002". In: *Journal of Marine Systems* 68.1-2 (2007), pp. 201–214.

[19] AK Mazur, Anna K Wåhlin, and Adam Krężel. "An object-based SAR image iceberg detection algorithm applied to the Amundsen Sea". In: *Remote Sensing of Environment* 189 (2017), pp. 67–83.

[20] Flavio Parmiggiani et al. "SAR analysis of the Larsen-C A-68 iceberg displacements". In: *International Journal of Remote Sensing* 39.18 (2018), pp. 5850–5858.

[21] Jennifer Salau and Joachim Krieter. "Instance segmentation with Mask R-CNN applied to loose-housed dairy cows in a multi-camera setting". In: *Animals* 10.12 (2020), p. 2402.

[22] MP Schodlok et al. "Weddell Sea iceberg drift: Five years of observations". In: *Journal of Geophysical Research: Oceans* 111.C6 (2006).

[23] JN Schwarz and MP Schodlok. "Impact of drifting icebergs on surface phytoplankton biomass in the Southern Ocean: Ocean colour remote sensing and in situ iceberg tracking". In: *Deep Sea Research Part I: Oceanographic Research Papers* 56.10 (2009), pp. 1727–1741.

[24] TAM Silva, GR Bigg, and KW Nicholls. "Contribution of giant icebergs to the Southern Ocean freshwater flux". In: *Journal of Geophysical Research: Oceans* 111.C3 (2006).

[25] Aidan Starr et al. "Antarctic icebergs reorganize ocean circulation during Pleistocene glacials". In: *Nature* 589.7841 (2021), pp. 236–241.

[26] Haroon Stephen and David G Long. "Study of iceberg B10A using scatterometer data". In: *IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No. 00CH37120)*. Vol. 3. IEEE. 2000, pp. 1340–1342.

[27] KM Stuart and DG Long. "Tracking large tabular icebergs using the SeaWinds Ku-band microwave scatterometer". In: *Deep Sea Research Part II: Topical Studies in Oceanography* 58.11-12 (2011), pp. 1285–1300.

[28] Ramon Torres et al. "GMES Sentinel-1 mission". In: *Remote sensing of environment* 120 (2012), pp. 9–24.

[29] USNIC. *Antarctic Iceberg Naming and Tracking Information*. 2022. URL: `https://usicecenter.gov/Resources/AntarcticIcebergs` (visited on 04/18/2022).

[30] Christine Wesche and Wolfgang Dierking. "Iceberg signatures and detection in SAR images in two test regions of the Weddell Sea, Antarctica". In: *Journal of Glaciology* 58.208 (2012), pp. 325–339.

[31] Christine Wesche and Wolfgang Dierking. "Near-coastal circum-Antarctic iceberg size distributions determined from Synthetic Aperture Radar images". In: *Remote Sensing of Environment* 156 (2015), pp. 561–569.

[32] Yucheng Zhang et al. "Recent advances of single-object tracking methods: A brief survey". In: *Neurocomputing* 455 (2021), pp. 1–11.