UiT
THE ARCTIC
UNIVERSITY
OF NORWAY

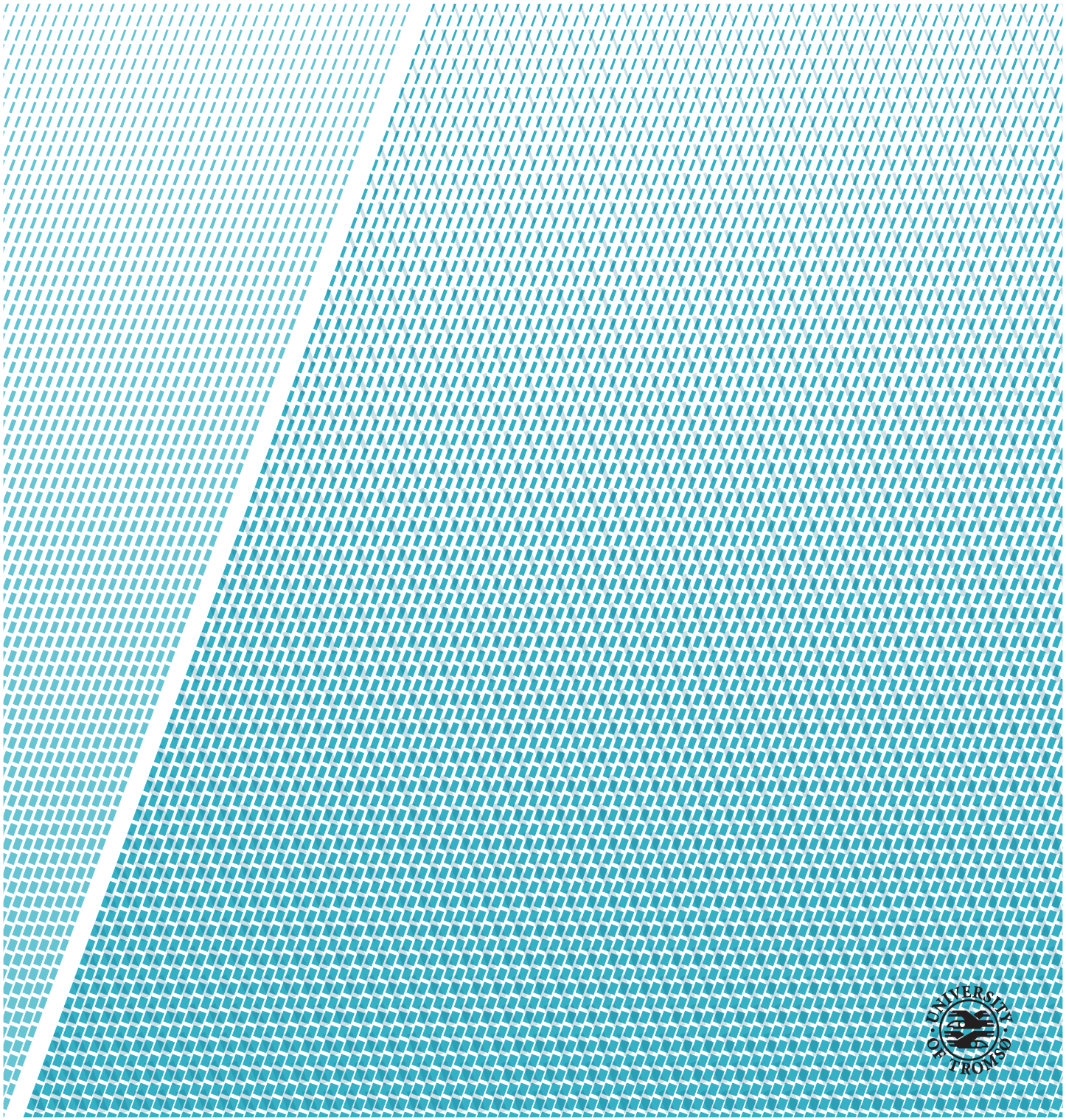
Faculty of Science and Technology
Department of Physics and Technology

An investigation of the robustness of distance measure-based supervised labelling of segmented remote sensing images

Åshild Kiærbech

FYS-3941 Master's thesis in applied physics and mathematics, 30 SP

May 2019



“ By the breath of God ice is given,
and the broad waters are frozen.
Stand still and consider
the wondrous works of God.”
–Job 37, 10. 14

Abstract

Unsupervised clustering methods on remote sensing images have shown good results. However, this type of machine learning needs additional labelling to be an end-to-end classification in the same manner as traditional supervised classification. The automation of the labelling needs further exploration. We want to investigate the robustness of a supervised automatic labelling scheme by comparing a segmentation with additional automatic labelling against a supervised classification method.

Using synthetic aperture radar (SAR) satellite images of sea ice from Sentinel-1, an automatic Expectation Maximization method with a Gaussian mixture model is used for the segmentation, taking into consideration the incidence angle variation within a SAR image. The additional labelling is a likelihood majority vote related to the Mahalanobis distance measure. The Bayesian Maximum Likelihood (ML) is used as the fully supervised reference method. The experiments of comparison are done using various amounts of training data and different percentages of mislabelling in the training data set. The classification results are compared both visually and using classification accuracy.

As training data size increases, the accuracy of the ML method tends to decay faster than for the segment-then-label approach, particularly when sample sizes per class are less than a hundred. As more contamination is introduced, the decay is not distinct, probably due to the large within-class variations in the training set.

Based on the results, the ML method generally gets a higher overall classification accuracy, but there are weak tendencies for the segment-then-label method to be more robust to decreasing training data size and more mislabelling.

Acknowledgements

First I would like to thank my supervisor Ass. Prof. Anthony Doulgeris for involving me into his projects and supervising me during the master's degree work. Secondly, I want to thank Johannes Lohse for providing me with labelled training data and for helping me with concepts and technical details.

Thanks go to my fellow students who have worked together with me during the years of study. I would also like to thank my father and mother for all they have given and are. A special thanksgiving goes to Henrik for existing and proofreading.

Lastly, I thank the almighty God, who created our beautiful Earth and the human brain, for all the great and astounding things that he has done for us.

Åshild Kiærbech
Tromsø, May 2019

Contents

List of Figures

List of Tables

Acronyms

EM Expectation-Maximization

EPFS Extended Polarimetric Feature Space

ESA European Space Agency

EW Extra Wide swath mode

GMM Gaussian Mixture Model

GRD Ground Range, Multi-Look, Detected level-1 product

GRDM Ground Range, Multi-Look, Detected Medium resolution level-1 product

MAP Maximum A Posteriori

ML Maximum Likelihood

MRF Markov Random Field

NESZ Noise Equivalent Sigma Zero

S1 Sentinel-1

SAR synthetic aperture radar

SNAP Sentinel Application Toolbox

WMO World Meteorological Organization

1

Introduction

Remote sensing of Arctic areas is an important scientific field used for the purpose of environmental and climate studies, marine traffic, and meteorology. Our focus is remote sensing and classification of sea ice, which is important for shipping purposes and for climate research. Of general interest is the extent, amount, and thickness of the ice in different seasons. Unfortunately, classification of sea ice is a field where only little ground truth (GT) data is available. To achieve GT data for all different types of ice in different seasons would be a cumbersome— not to mention expensive—process. The acquisition of new points of ground in situ data each year is limited.

There exists a lot of different physical sea ice types. The World Meteorological Organization (WMO) Sea ice nomunclature contains numerous different distinct ice classes (*WMO Sea-ice nomenclature*, 2017). The ice is partitioned depending on the development of the ice, the form of the ice, its concentration or frequency on the water, its origin, stages of melting, and so forth. For classification purposes, the different types must be limited and specified. E.g. Ochilov and Clausi (2012) use seven different classes in their sea ice classification.

Supervised classification methods, as we will see, are based on having loads of training data. For practical reasons, this is not a possibility in the Arctic areas. This makes the motivation and necessity for wisely utilizing the little training data at hand in the classification task.

1.1 Background on data classification

Supervised and unsupervised classification are two of the main sub fields of machine learning, both endeavouring to learn patterns within datasets for being able to categorize the data into subcategories. The supervised techniques have come far in the progress of satellite image analysis. Also, the unsupervised methods have now come far, and there are numerous ways of effectively sorting data. One of the upcoming challenges when using unsupervised techniques on remote sensing images is how to automatically determine the physical ground types that the classes represent. There may be large seasonal variations within the different ground types, and viewing geometry and sensor noise provide additional challenges.

Clustering, as an unsupervised machine learning method, can be used as a part of a classification process. To be called a classification, the subgroups need meaningful informative labels with known interpretation, not only random numbering entities. Therefore, the clustering needs a labelling on top, a process that determines what the constructed clusters are. The labelling may be done manually, but could possibly be automated, and that is the motivation of this study.

1.2 Previous work on segmentation and labelling

A survey on sea ice classification, presenting an overview on classification on sea ice based on SAR data, is presented by Zakhvatkina et al. (2019). The studies in the articles summarized below are mostly on sea ice, but a few are applied on land cover and agriculture. Some are more generally applicable on hyperspectral imagery and X-band SAR. Hyperspectral imagery is becoming an important research field as the satellite technology develops, but SAR is still the most applicable for Arctic conditions (see Chapter 3).

For classification in general, machine learning methods—like Gaussian Processes (Bazi and Melgani, 2010) and Neural Networks (Zakhvatkina et al., 2013; Maggiori et al., 2017; Ressel et al., 2016; Koltunov and Ben-Dor, 2001)—have been investigated, and these have also shown good performance for remote sensing imagery. Zakhvatkina et al. (2019) also list up Support Vector Machine and wavelet transforms, as well as Maximum Likelihood and Bayes classifier as possible sea ice algorithms that have been investigated and tested for sea ice. The latter two will be further discussed in this thesis. Now we will focus on the unsupervised part of sea ice classification, namely using segmentation and labelling.

1.2.1 Segmentation

Segmentation of remote sensing images is well investigated, also concerning sea ice. The clustering techniques in use vary according to the purposes. Among the algorithms are the well-used ISODATA (Parshakov et al., 2014*a*), as well as statistical histogram thresholding (Cutler et al., 2015), and other methods such as IGRS (Iterative region growing using semantics) (Yu and Clausi, 2008), and watershed algorithm (Soh et al., 2004; Ochilov and Clausi, 2012). Yu et al. (2012) use the segmentation algorithm MIRGS (Multivariate Iterative region growing using semantics) presented by Qin and Clausi (2010). Further, probabilistic clustering methods for mixture models, like Gaussian Mixture Model (GMM) (Koltunov and Ben-Dor, 2001) and Spectral Mixture Models (SMM) (Fang et al., 2018), are utilized. When using mixture models and class membership vectors, a hard or soft decision is still required. This gives room for misclassification where distributions overlap. Applying a Markov Random Field (MRF) smoothing helps correcting the pixel affiliations by considering the local spatial neighbourhoods for updating the cluster priors (Doulgeris, 2015; Fang et al., 2018).

1.2.2 Labelling

Reading the literature, one must be aware of the different terminologies used within classification. The word "label" is sometimes used meaning the class membership of a mixture (Fang et al., 2018). These "labels" are uninformative, in the sense that they are integers specifying the mixture component or cluster number. In this thesis "label" denotes only informative labels representing a physical meaning.

Regarding the labelling methods in the literature reviewed, many still do the process manually, using histograms and simple logic (Cutler et al., 2015). They are often most interested in the clustering task. For the purpose of getting cluster functionals, even a mathematical framework is proposed that uses threshold measure (Lyons and Arribas, 2018).

Another method used is approaching a more automatic way of labelling by dividing the image into polygons, where the polygons are the objects that are segmented. The segments are then labelled by utilizing the ice types and within-polygon concentrations from the polygons' egg codes, which are made by experts (Ochilov and Clausi, 2012).

ARKTOS is a rule-based system developed especially for classification of morphological image features, rather than pixel-wise classification. After the segmentation, attribute measurements for the segments are generated and passed

on to a rule-based classification. The segments are then labelled according to predefined expert system rules. (Soh et al., 2004).

Further, there are articles using distance measures for automatic labelling, e.g Mahalanobis distance when using a Gaussian mixture model (Moen et al., 2015), and Z-score distance for use in spectrogram comparison (Parshakov et al., 2014*a*,*b*). The difference between Moen et al. (2015) and Parshakov et al. (2014*a*) is that the former use training sample pixels extracted from the image, whereas the latter use training representations, which are pure reference endmembers in a library. The latter use hyperspectral data and the bands from the multispectral data as features, and the former use SAR data and SAR textural features. Distance measures are linked to probability distributions, that support the use of distance measures for comparing endmembers.

Ochilov and Clausi (2010) state the problem of doing labelling on top a segmentation for proper classification. Their objectives is automatic labelling of segmented sea ice images, and they test a combined segmentation and labelling process. IRGS is used as their segmentation algorithm. Even though this article focuses on automatic labelling, the segmentation still holds aspects of manuality for making the polygons in IGRS, and therefore the process is not fully automated.

Size and mislabelling

Gabrys and Petrakieva (2004) conduct experiments with different relative amounts of labelled data to unlabelled data. They find, not surprisingly, that a small amount of labelled data results in higher variability, and that a small amount of training data gives results with higher dependency on the reliability of the labelled data. Experiments where conducted to investigate the reliability of the labelled data, using three different ways of selecting labelled samples. Two ways of random sampling methods were tried, namely selecting randomly per class, making sure all classes were represented, and totally random, with the risk of some classes not being represented. The selective sampling methods where (1) the mean selection, rewarding samples close to the cluster mean, and with ability to split the cluster into subclusters; (2) the boundary selection, rewarding samples with highest distance from other samples with the same class; (3) a modification of (1) where clusters cannot be split. The selective sampling methods improved both the mean classifier performance and the reduction of the classification variance.

An empirical study on learning from both labelled and unlabelled data is done by Chawla and Karakoulas (2005). They investigate the use of additional unlabelled data together with the labelled data on both artificial and real datasets.

Multiple semi-supervised techniques for classification and one supervised technique are compared, using varying ratios of labelled to unlabelled data amounts, given by [(labelled, unlabelled)%], and different levels of contamination[(0, 5, 10, 20)%]. Contamination is mislabelling, or also called label noise. They find that some semi-supervised techniques perform better than the supervised technique for most datasets. The trend is especially strong when there are little training data and relatively much unlabelled data (1,99)%. Experimenting with mislabelling the datasets, they found that using 5% and 10% contamination with small relative amount of labelled data (1,99)%, semi-supervised methods performed better than the supervised method. The semi-supervised performed appreciable better also for the labelled/unlabelled percentages (10,90)% for 20% contamination.

This review shows that the field of labelling of segmented images has yet only been touched to a limited extent, and some researchers even concluded that *"limited research has been performed in ice-type labelling"* (Ochilov and Clausi, 2012, p. 4399).

1.3 Objectives

The scope of this thesis is to investigate and compare the performance of two automatic image classification schemes; (1) a semi-supervised scheme that does an unsupervised clustering in combination with an automatic supervised labelling (hereafter: segment-then-label), and (2) a fully supervised, or direct, classification method (hereafter: fully supervised).

A Gaussian mixture model within an Expectation-Maximization framework will be used as segmentation, and a distance-based labelling method will be used in combination with it. The Maximum Likelihood is chosen as the fully supervised method.

The main question is whether adding training data after a segmentation will give better results than a fully supervised classification where training data is provided from the beginning. We will compare the two approaches to see which is better when training data is limited, and when training data is contaminated.

The robustness of the two different main schemes are to be tested. If the supervised method performs better when the data is clean and enough data is present, how much contamination or how small sample size is sufficient for the segment-then-label scheme to be a better classifier?

The workflow is briefly described in the following steps:

1. Implement a Bayesian classifier to demonstrate the fully supervised method.

2. Use a ready-made segmentation algorithm to get segmentation results (Doulgeris and Cristea, 2018).

3. Implement a labelling strategy suggested from the literature for automatic labelling of the segments (Moen et al., 2015).

4. Compare the performance of the fully supervised method (1) and the segment-then-label method (2-3) with respect to the number of training data and contaminations.

1.4 Structure of the thesis

Chapter 2 is an introduction to the theoretical background for classification. Theory of machine learning concepts within supervised and unsupervised classification are described, giving an introduction to the scientific problem of choosing between supervised and unsupervised classification.

Chapter 3 describes the Sentinel-1 synthetic aperture radar images used in this thesis, and some challenges concerning this data.

Chapter 4 deals with the questions about the amount of data, its representativeness, and discusses seasonal changes of sea ice.

Chapter 5 explains the the preprocessing steps done to the Sentinel-1 GRDM products. Radiometric calibration, thermal noise removal, and multilooking are dealt with.

Chapter 6 contains information on the training data and how it is extracted from image polygons.

Chapter 7 define the Gaussian tubes—Gaussian functions with variable means, used for the methods in both chapter 8 and 9, to deal with the challenge of the class variation for incidence angle.

Chapter 8 present the Maximum Likelihood fully supervised classification method, and its implementation.

Chapter 9 present the segmentation-then-labelling method, and goes into the details of the segmentation and the labelling procedures.

Chapter 10 is on the comparison experiments of the two methods. The experiments are described and results presented and discussed.

Chapter 11 concludes the study, discusses strengths and limitations, and presents suggestions for future work.

Part I

Background theory and the image data

2

Machine learning theory

The field of machine learning consists of a number of subfields, where the two subfields of supervised and unsupervised learning are discussed here. The theory is important for understanding the problem of this thesis, and the methods explained later.

The difference between unsupervised, supervised and semi-supervised classification is described in many text books, such as Campbell and Wynne (2011, chap. 12.3-12.4) and Theodoridis and Koutroumbas (2009, chap 11). Reviews of basic models in unsupervised learning are found in Ghahramani (2004) and Friedman et al. (2001, chap 14).

Supervised classification of an image is based on having small sub-regions, or training sets, as reference for all the pixels to be classified. Based on the values of the sub-regions, the pixels in the image are assigned to their specified classes. If no training data is provided for a certain class, supervised algorithms are not able to recognize those certain pixel groups. The algorithm will require training data for all the classes. If training data is not available, better performance would be achieved by an unsupervised approach. Supervised classification are based on having labelled data for training the classifier. The trained classifier, or the decision lines with associated weights, are the basis of the class decision for the data points. The specific supervised method used in this thesis is contained in Section 8.1 on the Bayes classifier. The statistical background for this method is contained in Appendix A.

Unsupervised learning, or clustering, is the identification of natural groups or segments within the data, which are defined, identified, labelled, and in the end, mapped. Image segmentation uses no labelled training data, and group pixels together in segments, or clusters, based on their features and statistical proximity. This would lead to more natural groupings of pixels. The drawback is that clusters are unknown groupings, not categorized to a certain class. Clustering is investigated further in section 2.1. Note that the terms clustering is used interchangeably with segmenting in this thesis.

Semi-supervised classification is a hybrid category in between unsupervised learning and supervised classification. These methods are described in the textbooks (Theodoridis and Koutroumbas, 2009, chap 10), e.g. when using a supervised method with training data, in conduction with the distributions of the underlying structure of the data. A clustering is performed first without training data, before comparing clusters with additional training data for labelling the clusters. Semi-supervised methods are based on having training data, but not enough for doing it all supervised with a satisfactory result. The segment-then-label method we treat in this thesis is a type of semi-supervised classification, where we also want to test pushing the amount of labelled data used to a minimum.

2.1 Clustering

Clustering is based on dividing a set into more subsets, where a subset consists of "similar" elements, and is separated from other subsets due to some proximity criterion. **The cluster assumption** states that two points located in the same cluster are probably members of the same class. A clustering of the dataset X is the partitioning of X into M sets, $C_1, \ldots, C_M$. It is restricted to the following conditions:

- $C_i \neq \emptyset, i = 1, \ldots, m$
- $\cup_{i=1}^{m} C_i = X$
- $C_i \cap C_j = \emptyset, i \neq j, j = 1, \ldots, m$

I.e., each subset has to be non-empty, all elements in the dataset X are contained in some cluster, and each cluster is a disjoint region separated from other clusters.

The third definition restricts each data point to belong to only one cluster. By introducing a **membership function**, a point can temporarily be affiliated

to different clusters in a probabilistic manner. A membership function is the mixing portion specifying the probability for a data point to belong to each cluster. In other words, how much each cluster contributes to the pixel mixing. A dataset X is partitioned into m clusters, where the membership function u_j is the membership in jth cluster for each point. u_j is contained in the inclusive interval:

$$u_j : X \rightarrow [0, 1], \quad j = 1, ..., m$$

such that the fractions for one data point sum up to 1

$$0 < \sum_{j=1}^{m} u_j(\mathrm{x}_i) < 1 \quad j = 1, ..., N \quad j = 1, ..., m$$

and such that each fraction is at maximum 1

$$0 < \sum_{i=1}^{N} u_j(\mathrm{x}_i) < N, \quad j = 1, ..., N \quad j = 1, ..., m$$

2.1.1 Proximity measures

Clustering algorithms based on pixel similarity need a proximity measure for quantifying how similar, or dissmilar, the pixels are. A dissimilarity measure is often called a distance, due to the proportionality between distance and dissimilarity. (Theodoridis and Koutroumbas, 2009, chap 11.2). For probabilistic clustering schemes like the Expectation-Maximization, similarity is measured by likelihoods. Also for the supervised Maximum-Likelihood the likelihood is used as a measure of similarity. In the *labelling* stage, some proximity measure is needed for calculating the proximity between a cluster and training data points or clusters.

An example of a dissimilarity measure is the Mahalanobis distance given by

$$d(\mathrm{x}, \mathrm{y}) = \sqrt{(\mathrm{x} - \mathrm{y})^T \Sigma^{-1} (\mathrm{x} - \mathrm{y})}$$

This is the close to the Euclidean distance, the difference being that the Mahalanobis distance assumes some feature covariance. A Mahalanobis-based distance clustering is equivalent to a probabilistic Gaussian Mixture Model (see section 2.1.2).

Proximity measures are used to find the distance between single data points in a vector-space, but also for finding proximity between a single point and a set, for possibly assigning the point to the set. Two ways of comparing a point to a cluster are:

- all points in the set contribute to the proximity, using the maximum, minimum, or average proximity function.

- or; the proximity is between a representative for the set and the point. The representative for a compact cluster is a point, and the representative for a Gaussian mixture component is described by both a mean point and a covariance.

2.1.2 Mixture models

Different algorithms with a large variety are developed for the purpose of dataset clustering. In this thesis, we focus on a cost function optimization-based clustering. The particular algorithm we are looking at is the Expectation-Maximization (EM) algorithm utilizing a GMM. The EM algorithm is briefly discussed in Chapter 9, but we want to introduce the GMM here.

Using a mixture model, a point is still belonging to one cluster only. As this cluster is not yet known, a membership function is utilized for determining the likelihood of a point belonging to the different possible clusters. The mixture model is written as the sum of all model components weighted by their importance. The distribution of the data points in a mixture model of K components may be written as

$$p(\mathrm{y}|\theta) = \sum_{k=1}^{K} \pi_k p(\mathrm{y}|\theta_k)$$

where π_k is the class memberships of component k such that

$$\sum_{k=1}^{K} \pi_k = 1 \quad \text{and} \quad \pi_k > 0 \quad \forall k$$

and $p(\mathrm{y}|\theta_k)$ is the probability density function for the random variable y, with parameters θ_k.

The question in a classification task is how to find the mixture components contained in the data. The EM algorithm is a possible option that we will discuss later.

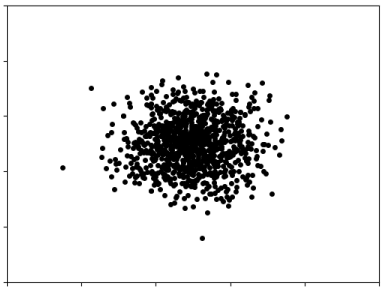


Figure 2.1: A compact cluster with points centred around a centre point.

Gaussian mixture models

A Gaussian Mixture Model (GMM) is a simple mixture model, where the distribution of a component Y is assumed to be normal:

$$Y = \mathcal{N}(\mu, \Sigma)$$

The parameters for the model of component k thus become $\theta_k = (\mu_k, \Sigma_k)$, where μ_k is the mean vector, and Σ_k is the covariance matrix of the kth Gaussian component. The Gaussian mixture model assumes the clusters to be compact, having a certain variance around a mean point, see Figure 2.1.

3

The satellite images

This thesis is a work on image classification, in the first place being a comparison of two machine learning algorithms. However, this is done for a certain type of images. Their distinctiveness makes it necessary to describe them and their certainties. The images are the Ground Range, Multi-Look, Detected level-1 product (GRD) from the Sentinel-1 (S1) spaceborne synthetic aperture radar (SAR) satellite. We will briefly discuss the nature of these images in the following sections. Two major problems dealing with classification of S1 images are discussed in Sections 3.6 and 3.7. In the end of the chapter we discuss feature selection for SAR images.

3.1 Sentinel-1

The S1 satellite is part of the European Space Agency (ESA)'s Sentinel family. It operates with C-band SAR having a frequency range within the microwave region (central frequency of 5.404 GHz). Because of its capability for wide coverage (400 km wide areas), its resolution, and its short revisit time, it is used among other applications for maritime surveillance of the large ocean areas. Among the sensor's four different acquisition modes is the Extra Wide swath mode (EW), used for the imaging of large arctic and marine areas—especially applied within ocean monitoring for crude oil detection, ship detection, and sea ice monitoring. The EW mode has five imaging swaths in the range direction (Collecte Localisation Satellites (CLS), ESA, 2016). Alongside the Single Look

Complex (SLC) level-1 product and Ocean (OCN) level-2 products, the GRD product is one of the S1 data products distributed by ESA. The S1 images are free and openly available.[1]

3.2 The synthetic aperture radar

The S1 is a synthetic aperture radar (SAR) sensor. This brief introduction to its theory is based on the more thorough reviews found in Elachi and Van Zyl (2006, ch. 5-6) and Campbell and Wynne (2011, ch. 7).

As the SAR sensor operates with *microwaves* it has certain advantages. The wavelength of microwaves is larger than the atmospheric particles, and therefore the wave propagates unhindered trough the atmosphere. The signal is also unaffected by different weather conditions and lack of Sun illumination. This makes it suitable for Arctic areas with tough weather conditions, and darkness through the winter season.

Scattering occurs for wavelengths in the microwave and radio part of the electro-magnetic spectrum. This phenomenon happens when the signal waves interact with the target surface, and the wave is reflected in some direction. The *backscatter* is the portion that is scattered back towards the satellite and detected by the sensor. Depending on the wavelength, intensity, polarisation, phase, and other properties of the radiation, we can say something about the surface. Different surfaces, like various types of sea ice and water, will scatter differently based on the surface scatter type, meaning the surface's conductive properties and geometrical structures.

The SAR is an *active sensor*, meaning it both transmits and receives signals, as opposed to the passive ones that only receive. This nature of the active microwave sensor makes is possible to compare the transmitted and received signals, in order to get more precise information of the ground surface. The **backscatter cross section** is defined as the ratio of backscatter signal from ground to sensor over the transmitted signal from the sensor. This ratio is an indication of the particularities of the surface, e.g., how the ground material absorbs or scatters in other directions. The microwave sensor produces gray-scale images of the surface backscatter.

Speckle is a SAR image effect, which appears as salt-and-pepper noise (bright

1. The Sentinal-1 scenes were aquired during the European Space Agency's @Copernicus Programme. For information on image access, see: `https://sentinel.esa.int/web/sentinel/sentinel-data-access`

and dark return values) in the image (Campbell and Wynne, 2011, p. 222). The SAR signal is transmitted over a narrow range of wavelengths, and is a coherent source signal containing both an amplitude and a phase part. Random displacement of individual scatterers causes constructive and destructive interference in the coherent signal, resulting in the scattered energy to be either reinforced or suppressed. Speckle noise is multiplicative, meaning that it is directly proportional to the radiance of the specific pixels in the image, and is an intrinsic part of the signal.

3.2.1 Polarimetry

An electro-magnetic wave consists of coupled electrical and magnetic fields. The two fields are orthogonal to each other and to the propagation direction. The amplitude of the electric field is a function of its orthogonal polarisations (Canada Centre for Mapping and Earth Observation, Natural Resources Canada, 2015).

Depending on the antenna configuration, it may transmit and receive in horizontal (H) or vertical (V) linear polarizations. Configured with single-polarisation, it receives only in the polarization it transmits (giving channels HH or VV). If dual-polarisation is provided, it transmits in either H or V, and is able to receive in both polarizations (giving channels HH/HV or VV/VH). With quad-polarisation, it transmits and receives both polarisations (channels HH/HV/VH/VV). Using dual-polarisation and quad-polarisation is beneficial as it gives the possibility to analyse backscatter in different channels simultaneously. Different surface types may have similar response in one channel and different response in another. Because of the certain polarimetric behaviours of the different surface types, the polarimetry may be of great use in a classification task.

The S1 EW mode is available in single and dual polarisation. The dual horizontal transmitted configuration (HH/HV) is found to be the best suitable in marine polar areas and to improve the sea ice monitoring (Copernicus Space Component Mission Management Team, 2018, see p. 14,41,49). In sum, for our classification task, we get two polarisation channels (HH and HV), corresponding to two feature layers of different distinguishing capabilities.

3.3 Multilooking

Image multilooking is smoothing the image, or averaging the values in a pixel neighbourhood. This can be done in the Fourier domain, by splitting the Fourier

transformed image to the wanted number of looks and then average over these. Alternatively, it can be done with a running average filter in the spatial domain. The latter is used for multilooking in this thesis.

The averaging of image values results in reduced speckle and thermal sensor noise. The averaging must be done on the real intensity values. Averaging over complex zero-mean values, results in averaging to zero, instead of averaging to the non-zero mean-intensity.

3.4 Resolution and pixelspacing

The GRD is the focused SAR data that has been multilooked and projected to ground range. We use the medium resolution, thus GRDM, but high resolution is also available for the EW mode's GRD product. The product is multilooked with six looks in ground range (rng) direction and two looks in azimuth (az) direction. It's pixel spacing is 40x40 [rng x az] [m], which means one pixel correspond to an area on ground of 40m x 40m. The resolution[2], or resolving power, though, is 93x87 [rng x az] [m]. Here the resolution is different from the pixel spacing, which tells us that the image is already blurred over the pixels it holds. The resolution tells how far apart two distinguishable objects on the surface are. Ice types with an extent of more than the resolution size can be distinguished.

3.5 SAR applied on sea ice

According to Elachi and Van Zyl (2006, p. 172), "*one of the most useful applications of spaceborne microwave radiometry for surface studies is in the mapping of polar ice cover and monitoring its temporal changes*". In this section we will briefly discuss the SAR measurements and the SAR feature selection typical for sea ice.

The sea ice scattering is dependent on surface roughness, size of scatterers inside the ice, and its dielectric properties. The latter are in turn dependent on the local temperature and salinity, as the salt molecules in the ice reduce the radar penetration (Haykin et al., 1994). *Volume scattering* is significant for some ice types, e.g. multi-year sea ice, which in general have lower salinity. For other ice types, e.g. first-year ice, *surface scattering* dominates (Onstott

2. The resolution corresponds to the mid range value at mid orbit altitude, averaged over all swaths.

and Shuchman, 2004, p. 87, 89). The difference in dielectric properties of the open water and the sea ice makes the backscatter of the two considerably different.

3.5.1 Feature selection

A better classification can be achieved by a reasonable feature selection, using the input features with the best class distinction capabilities. Different features may highlight different surface targets, and certain feature combinations may give higher ability to distinguish classes. Using more than one polarisation channel gives the opportunity to create new features based on combinations of the channels. Some examples of feature selection applicable for sea ice classification in remote sensing images are worth mentioning.

The Extended Polarimetric Feature Space (EPFS) contains six features; one for non-Gaussianity and five polarimetric ones from the covariance matrix. It utilizes the advantages of quad-polarimetry. The texture and polarization features hold the geometric brightness distinctive for SAR. (Doulgeris and Eltoft, 2010; Moen et al., 2015)

For sea ice/water distinction, Scheuchl et al. (2001) have found the HV-intensity, the HH/VV ratio, and the anisotropy to have good distinction capabilities. Zakhvatkina et al. (2013) found that the most informative texture features for distinguishing some specific ice types (MYI, FYI, DFYI, LFYI and open water)[3] are correlation, inertia, cluster prominence, energy, homogeneity, and entropy, along with the third and fourth central moments of image brightness. Ressel et al. (2016) discuss polarimetric features for X-band SAR, and use both the complex backscatter, the H/A/α, and eight more features related to texture.

For the simplicity of this study we restrict the number of features to the intensity values for the horizontal transmitted dual-polarisation's two channels.

3.6 The incidence angle problem

A side-looking sensor looks with different incidence angles on the areas on ground within one scene. This large incident angle range is particularly present in wide swath scenes. The Sentinel-1 (S1) EW mode has an incidence angle

3. MYI: Multiyear Ice, FYI: First-Year Ice, DFYI: Deformed First-Year Ice, LFYI: Level First-Year Ice

range of 18.9° – 47.0°. The radar backscatter tend to be (close to) a linear function of incidence angle, and the slopes vary with surface type. The intensity-incidence angle slope tends to be less inclined the more deformed the ice is, and steeper the higher the moisture content is (Mäkynen et al., 2002). The slopes will be different for the HH and HV polarisation channels. The non-constant incidence angle causes the same ground types to be clustered to different clusters if the incident angle difference between the ground type locations is too large.

Mäkynen and Karvonen (2017) review the research done on the front of incidence angle correction in sea ice classification and clustering. They experiment and find that the backscatter versus incidence angle slopes for S1 EW SAR. These slopes can also be utilized for other C-band SAR (e.g. RADARSAT-2), but only for the HH band, as the noise floor problem causes the slopes made for the HV band in S1 to be S1 specific. They also state that the incidence angle slopes change with seasonal variations.

Our way to deal with the incident angle problem is to make Gaussian tubes, Gaussian curves of angle dependent means and constant variation. An explanation of this is found in Chapter 7.

3.7 The noise floor problem

The satellite sensor needs a sufficiently strong signal to be able to record it, and the signal needs to be stronger than the Noise Equivalent Sigma Zero (NESZ) to be distinguished from the background noise. NESZ is a system parameter measuring the sensors sensitivity and calculated from the optimized antenna pattern. It is dependent on the antenna gain and efficiency, and relates to the Signal to Noise Ratio (SNR).

Thermal properties of the sensor artificially cause the measured response in each imaging swath to concentrate around the middle of the swath. This causes a within-swath variation, giving the effect of different pixel value levels from the middle part of a swath to the outer part of a swath. This is called the *noise floor problem*. Thermal noise occurs in both range and azimuth directions, and is seen as bright scalloping areas in the image. Especially the cross-polarisation channels are exposed to the noise floor problem, having generally lower backscatter cross section, but the same NESZ level (i.e. the SNR is lower). Also ground areas with low backscatter, like calm seas, are typically more affected by this problem.

The noise floor problem is a hinderance in a classification process, as the pixel

intensity value-based classification will misclassify the noise areas. To solve this problem, the noise has to be corrected for. The Sentinel Application Toolbox (SNAP) has a thermal denoising function, which procedure is described by Sentinel-1 Mission Performance Centre (MPC) (2017*b*). This function corrects for the noise to some extent, but cannot fully compensate for it. The problem and its solution so far is discussed by Park et al. (2018). As our study is limited, the noise floor problem is not investigated further in this thesis. To exclude this topic from our problem, the areas between the swaths are masked out, and only the mid-swath areas are used in the segmentation, training, and classification.

3.7.1 Swath emission

The Sentinel-1 EW images consist of five swaths. In addition to the noise between the swaths, the large difference between the image brightness for the first swaths compared to the others, should be carefully considered. In addition to masking out the noise between borders, one should consider if the first swath also should be masked out, due to both brightness and noise. By masking away the first swath, the classifier would then be training on the second to the fifth swaths only and the classification done for the same area. Unfortunately, a smaller part of the image would then be analysed. In this thesis we stick to use all swaths, to see how well the algorithms will work for the whole image range.

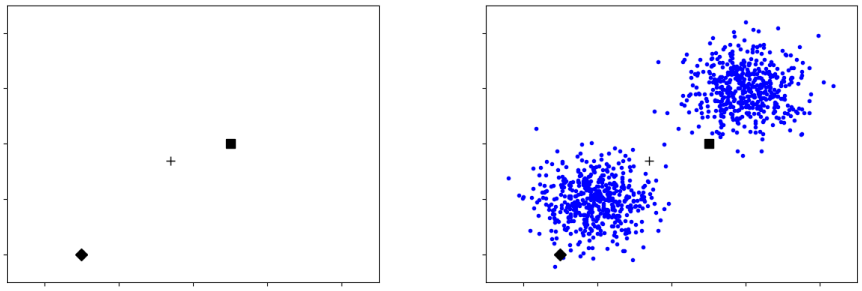
4

Data size and representativeness—including seasonal changes

The lack of—or limited amount of—labelled training data is what makes the clustering as an unsupervised method attractive, especially for remote sensing. In this chapter we will discuss the amount and representativeness of the data used for the labelling task. We will discuss the sensitivity to contamination in the training data, and the size of the training data. Included is also a section about seasonal changes.

4.1 Respresentativeness

The segment-then-label method has some challenges. Ideally, the method should correctly label the automatic generated image segments, based on training data. A general problem, however, is that the training data may not be representative. Putting it to the extreme: What if there only is one point per class, can we rely on the information of that point? An example is shown in



(a) **(b)**

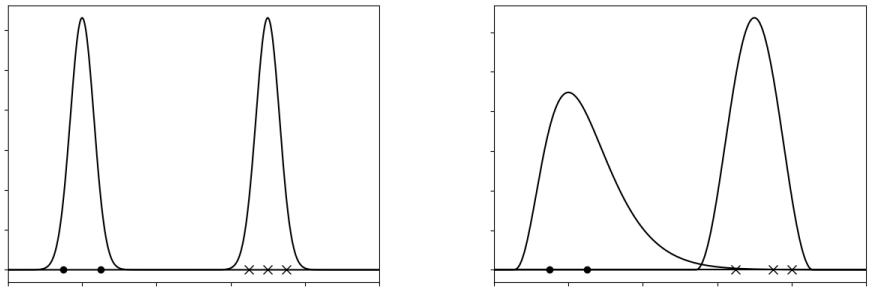
Figure 4.1: Given training data for two classes (square and diamond), which class would the new data point "+" belong to? Alternative (a) with only labelled training data points available. Alternative (b) including the data set's distribution. Figure from pilot.

the Figure 4.1a. Based on the training data alone, the new data point, marked as a cross, has the closest distance to the square marker, and would therefore belong to the same class as the square. Taking a closer look on the data set's distribution in Figure 4.1b, it is more reasonable to put it in the same cluster as the diamond marker. This illustrates why clustering would help solving the classification problem, but is also shows the importance of representative and reliable training data. In this case, both training samples are representing outlier values of each distribution.

Ideally, the training data should represent the whole range of possibilities. This leads to the next example explained by Figure 4.2. If the calculated distributions are based on the present training data only (given as dot and x), the distributions would probably look like in Figure 4.2a, yet, in reality, the distributions may look like in Figure 4.2b. The few data points do not fully describe the shape and location of the real distributions, being a source of possible classification errors. This is important to keep in mind, when *assuming* the Gaussian distribution in the Gaussian mixture model.

4.2 Overlapping distributions

Overlaps between the distributions lead to another problem: How to distinguish between overlapping distributions? Considering the supervised Bayes classification, the decision boundary is set such that the most probable class is chosen (see section 8.1). In the EM-algorithm (see section 9.1.2), assumptions on the underlying distributions are made in order to cluster the data according



(a) **(b)**

Figure 4.2: (a) Sample distributions based on present data points. (b) Real data distributions. The few data points are not able to fully describe the shapes and locations of the real distributions. Figure from pilot.

to both labelled training data and unknown membership data. However, both methods may be prone to failure when different data distributions overlap. In Figure 4.3, a distribution of a given data set is shown in red, with the true underlying distributions in black. It is clear by investigating only the red curve, that it may consist of at least three distributions. How could a new data point (the cross marker) be classified correctly? The data point could belong to either of the distributions, and thus the final decision could possibly be wrong.

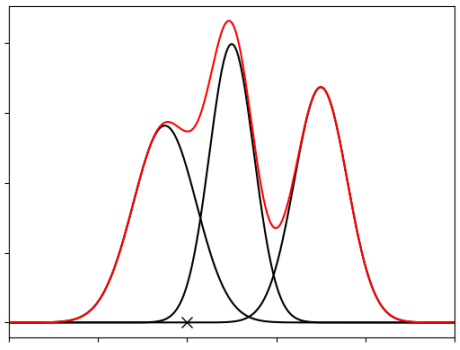


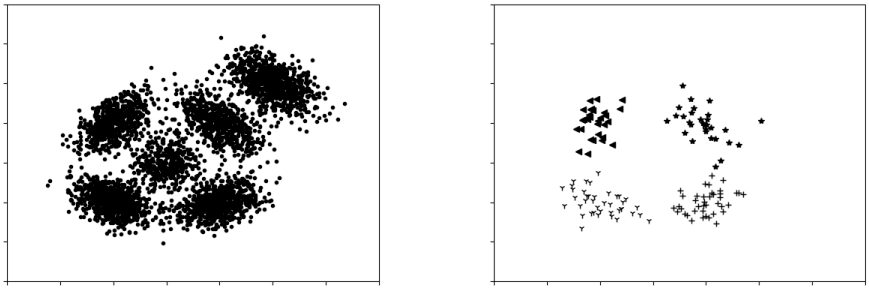
Figure 4.3: A Gaussian mixture distribution. The red line is the total distribution, consisting of the three black distributions. The new point (x) belongs to only one distribution, but how is the correct distribution determined? Figure from pilot.

4.3 Unknown data classes

Another question is whether a method is able to identify unknown data classes. As for sea ice, there are a lot of different ice types. If training data does not exist for a certain class within the image, the ideal method is still able to differentiate between the unknown cluster and the other labelled classes. An "unknown" label would specify that the method was not able to recognize a cluster's physical interpretable class, e.g. because the location of nearest possible label was too far away, or above a certain threshold.

Figure 4.4 illustrates a training data set that is missing data for some of the classes. The data set in Figure 4.4a seemingly contains six differentiable clusters. In the training data set in Figure 4.4b, only four of the six clusters are represented. Supervised classification of the data (Figure 4.4a) based on this training data (Figure 4.4b) yields only four classes, and the data points belonging to the missing classes will be classified into the four known classes. An unsupervised algorithm would identify six clusters, but the subsequent labelling algorithm would only be capable of labelling the data into four classes. The ideal algorithm, however, should be able to label the last two clusters as "unknown".

The essence of this chapter so far is that the resulting labelling is dependent on the training data at hand. A robust method is ideally less sensitive to small, incomplete, or biased training data.



(a) **(b)**

Figure 4.4: (a) The dataset consisting of six possible clusters. (b) Training data corresponding to only four of the clusters in (a). Figure from pilot.

4.4 Error sources related to remote sensing

In addition to the above mentioned sources of classification error, three sources directly relating to remote sensing imagery are worth pointing out.

Calibration is important for reducing noise from the satellite sensor and from the geometrical viewing conditions. Unfortunately, the calibration in itself is prone to error. Miscalibration may cause shifted class representatives, and thus misleading classes in the classification. In this work we do the calibration as described in 5, being aware of the potential error.

Incidence angle is a geometrical viewing condition which is another source of classification error, as discussed in Section 3.6. Different incidence angles for the actual data will cause mean intensity shifts, as illustrated in Figure 4.5. In the figure, all classes have shifted the same amount per angle and stays in the same class order. In reality the classes shift with a class-specific amount per incidence angle. The solution we use for this problem is explained further in Chapter 7 about the Gaussian incidence angle dependent tubes.

4.4.1 Seasonal variations of sea ice

Negligence of the seasonal changes of sea ice can result in the last type of classification error source that we will discuss in this chapter.

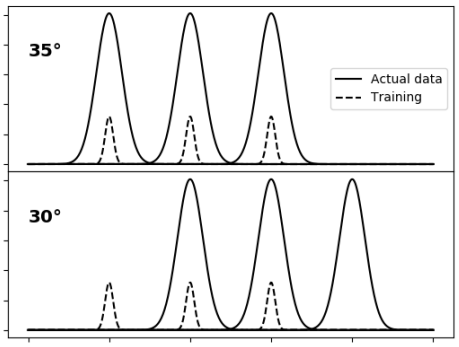


Figure 4.5: Incidence angle causes a shift in the actual data. Note that the cluster distributions (—) may have interchanged place, as the incidence angle slope per class may be rather different. The training data classes (- - -) may correspond to different clusters for 30° and 35° incidence angle. Figure from pilot.

Sea ice is a changing matter, which is worth keeping in mind when doing classification. The backscatter from an ice type may behave totally different from one season to another, such that using training data from one season for classification on an image from another season may be prone to error.

The sea ice's annual seasons are often referred to as being in different thermodynamic stages: freeze-up, winter, early melt, melt onset, and advanced melt (Mäkynen and Karvonen, 2017; Barber et al., 2001). The thermodynamic properties, along with the physical behaviour and dielectric properties of the ice, vary with the ice stages throughout the annual cycle. The different sea ice types will have different types of variations and changes when going from one season to the next (Barber et al., 2001). From the classification point of view, this means that classes being easily distinguishable in one season, are not necessarily distinguishable in another season.

The salinity and the snow pack will have impact on the backscatter. Studying at the micro plane, the pores and ponds in the snow will be able to fill themselves with either water or air, and thus give rise to different scattering mechanisms. The gradual deformation of the snow crystals also effects the backscatter. The amount of water in the ice or snow will cause a variation in dielectric properties, and thus give variation in the backscatter signal. Snow-covered sea ice will give different dielectric response, and this will vary with the amount of water contained in the snow. Both salt from the underlying sea and the solar illumination will give physical changes of the ice, resulting in the ice having different backscatter conditions (Barber et al., 2001).

For the melt and summer seasons the different sea ice classes will give more similar backscatter, as the surface conditions of the ice classes are more equal. Meltwater on the ice surface prevents the distinct ice surfaces below from being detected, due to less penetration. E.g. distinguishing flat ice from small ridges can be hard with surface meltwater. During the winter season, on the other hand, the ice is more stable, the ice types' backscatter levels being more different (Barber et al., 2001).

Many studies are done to investigate how to do remote sensing of the changing ice, thereby Park et al. (2016), who have investigated sea ice for the late summer and early autumn with melting and freezing periods, resulting into what can lead to better sea ice classification for those seasons where melt ponds occur on the ice and the ice is covered by a various amount of water, and ice properties are not stable. Another study (Casey et al., 2016) is on how to use both L-band and C-band to get better separability of multiyear ice (MYI) and first-year ice (FYI) in the season where the distinguishability in C-band is rather poor.

The overlap between the ice type's distributions vary across the year—i.e. one

ice type may shift its overlap with different other ice types during the year. The distribution of a specific season also may vary a lot from year to year, and for the different Arctic regions. The winter (typically at least January - April) is the season with least changing classes. One year May could be included in the winter season, whereas the melt season may begin during May. Melt, summer, and freeze-up stages have larger within-class changes. June can be the period of melt onset, and July and August will most often have summer season behaviour, before freeze onset in late August or September (Bliss et al., 2019).

With this background the training and test data should be carefully chosen. In our work we have training data polygons drawn for images acquired from March to July, both winter, melt, and summer season. We chose to use all these images even though they are from different seasons, as the error is likely to affect both our methods equally.

5 Preprocessing

The image products from the S1 satellite that we use are described in Chapter 3. An overview over the exact scenes used can be found in Tabel C.1 in Appendix C. These are used for training of the fully supervised classifier, and for the labelling stage within the segment-then-label. In this chapter we focus on the preprocessing of these scenes.

The Sentinel Application Platform, SNAP, is used for the thermal correction and radiometric correction preprocessing steps.

5.1 Thermal noise removal

The thermal noise comes from the properties of the sensor as described in Chapter 3.7, and can be removed or reduced. The NESZ thermal noise pattern in range direction for the S1 EW mode follows a specific pattern Sentinel-1 Mission Performance Centre (MPC) (2017*b*, p. 4 and 6). Based on this pattern a denoising vector for the range direction can be made Sentinel-1 Mission Performance Centre (MPC) (2017*b*, p. 11), which is then used in the denoising process. Similarly process is done for the thermal noise in the azimuth direction. Some of the thermal noise, especially in range direction, may still be present in the image after thermal denoising, leading to the noise floor problem discussed in Section 3.7. This occurs mostly in the cross-polarisation channel.

5.2 Radiometric calibration

A raw SAR image is impaired due to antenna gain and antenna effective area. Radiometric calibration of the SAr image is done to remove the image's dependency on the imaging sensor, and to adjust the image due to the geometrical viewing conditions; the incidence angle and the topographic conditions. After calibration, the image is independent also to the distance between radar and target.

The radiometric calibration ensures the possibility to compare geophysical variables derived from different points of time and from different sensors.

When doing the calibration, the user decides the projection of the image, either ground range or slant range. The GRD is projected to ground range, and the pixel values are the detected magnitudes. (Sentinel-1 Mission Performance Centre (MPC), 2017*a*, see Section 7.3.1)

5.3 Exported layers

After the thermal noise correction and the radiometric calibration are performed in SNAP, the intensity value images are exported as separate for further use. From SNAP are also retrieved the land mask, masking out land, and the incidence angle for the whole image. The land mask is then further modified, to ensure the mask covers both land and the nearest approximately 50 pixels to land. This is done by applying a convolution filter, via the Fourier domain.

Examples of the incidence angle image and the land mask image are illustrated in Figure 5.1.

5.4 Multilooking

The S1 Ground Range, Multi-Look, Detected Medium resolution level-1 product (GRDM) is already multilooked, see Section 3.3. Applying even more looks could nevertheless be a handy tool for our further image analysis.

Using more looks reduces speckle and noise variance, and consequently causes a higher radiometric resolution and reduction of class overlaps. On the other hand, the spatial resolution is decreased. Small areas, like single leads and ridges, may not be detected as such, because of the smoothing. Larger areas containing many of these may still be detectable, but then as mixtures of varying

abundances rather than being identified individually. Such mixture pixels have a higher abundance for this narrow ice class. The size of the mask has an impact on this, as more looks means more smoothing, thus more mixing within a single pixel's value. Smoothing the edges between the different ground targets also results in mixed pixels.

The number of looks is also related to the thermal noise patterns. If fewer looks are applied, the signal noise variance is higher, such that the thermal noise patterns are negligible. Thus the images are less affected of the thermal noise. By using more looks, the image noise is more reduced, the signal has less variability, and the thermal noise, which is no longer negligible, will have a larger influence on the image.

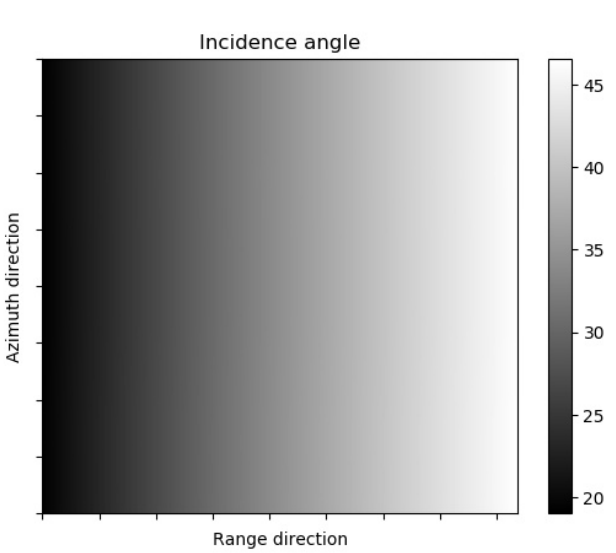
The GRD data, without any additional multilooking, suits for detection of ice areas larger than 40x40 m. If the goal is to do a close investigation of the ice, fewer looks may be better when using the EW mode. On the other hand, because the EW mode is so wide, it is well suited for monitoring wide areas. Thus it is applicable for large-scale projects, e.g. making larger-scale sea ice maps.

Figure 5.1c shows an example of a sub-image without any additional looks, and Figure 5.1d shows the same sub-image multilooked with a 5x5 sliding average filter. The appearance in the latter is clearly more blurred that in the first, but the general contours are still visible.

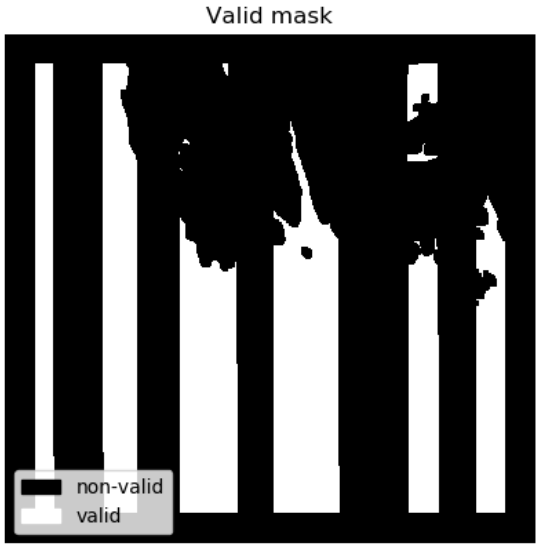
5.4.1 Downsampling

The original images have a size of ca. 10 000 x 10 000 pixels. The repetetive classification of many such images is a time-consuming process, thus the images are downsampled, every 5th pixel in each spatial direction being picked, in analogy to the multilooking process done with a 5x5 filter. This results in a downsampled image of ca. 2 000 x 2 000 pixels, a pixel amount that is one-twentyfifth of the original.

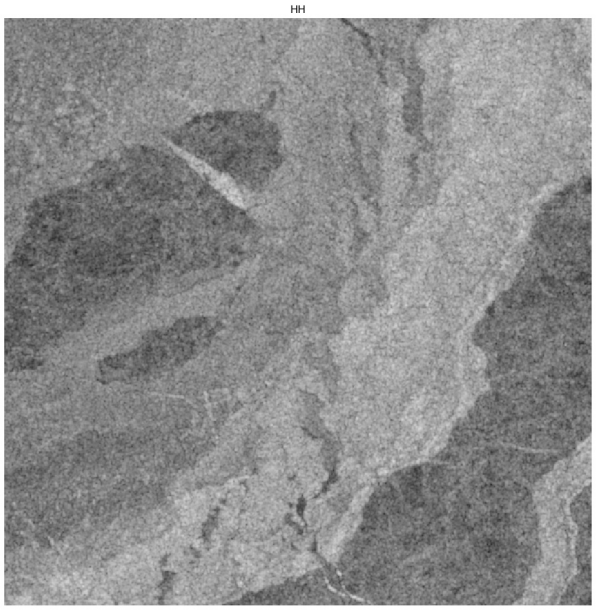
The downsampling changes the pixel spacing. (1) If no downsampling, the pixel spacing is kept the same as for the original image (40 x 40)[m x m], and the resolution is lowered from (93 x 87)[m x m] to (93 x L) x (87 x L). (2) If the image is downsampled with a step of (L x L), i.e. taking every Lth pixel in both spatial directions, the resolution is still lowered to (93 x L) x (87 x L), but the pixel spacing is now (40 x 40) x (L x L)=(40 x L) x (40 x L).



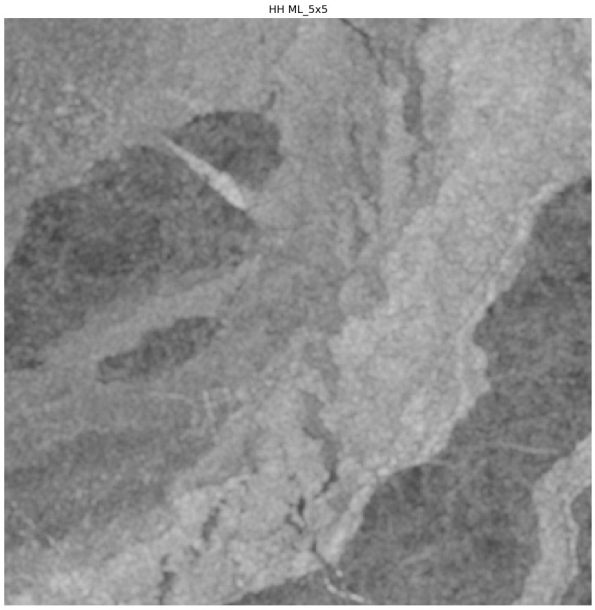
(a)



(b)



(c)



(d)

Figure 5.1: Examples of (a) an incidence angle image and (b) a land mask extracted using SNAP. The borders and between-swaths are masked away afterwards. Sub-images extracted from Image no. 2 (c) without additional multilooks and (d) with 5x5 looks [rng x az].

6

Training data and polygons

The data is as described in Chapter 3 and preprocessed as in Chapter 5. In this chapter we briefly describe the ice classes and the training data used in this thesis. A section about why the open water class was split in two is also included.

The exact scenes used in this thesis are listed in Table C.1 in Appendix C. These are chosen as it for these images existed training data polygons with different ice type examples (defined by J. Lohse, 2019)[1]. These are from the period from March-July, and we are aware that this is a possible source of error, see Section 4.4.1.

6.1 Polygons

When drawing the polygons, it is important that a broad range of incidence angles are covered for each class, for the purpose of getting more credible slope values. Therefore, ideally, as many images as possible should be included, in order to get enough training data for all classes, including the whole incidence angle range from a variety of ice type constellations. When joining training data from more images, one must ensure the polygons represent the whole incidence angle range. Otherwise there is a risk of having little incidence angle

1. The polygons for classes 1-5 are defined and drawn by J. Lohse, 2019.

variations within a class, making the slope of the class inaccurate. It is possible to classify an image based on the polygons drawn within its own frames. This is not recommended as the polygons may not spread over the incidence angle range, but one class may be concentrated in a specific part of the image. The training is not very time-consuming, taking a couple of minutes, as the training set are of limited sample size.

Polygons of the seven ice classes are drawn in 27 images (see Figure 6.1). The classes 6-7 are self-drawn.

6.2 Ice classes and training data

In this work seven ice classes are used, as they are clearly present in some or more of the images. By further investigation, these classes could have been divided into additional subclasses. The seven ice classes and their associated class number are listed in Table 6.1. The training data for theses classes are extracted from the mentioned image polygons, which are illustrated in Figure 6.1.

The training data amount for each class for each image is presented in Table C.2 in Appendix C. The particularly disturbed zones between the swaths are masked away from the images (see Figure 5.1b), such that only the mid-swath points are used for the analyses. By masking away these areas, we avoid the images being too much affected by the noise floor problem (see Section 3.7), which we want to ignore in this work. This is done as the incidence angle dependency in the data is still there.

Both from the polygons in Figure 6.1 and from Table C.1, we see that two classes (6 and 7) dominate the training data. This requires a careful subsampling from the classes, ensuring all classes are still well represented.

Data from the polygons in all images are combined to a common training data pool, where each sample consists of a HH value, a HV value, and an incidence angle value. Scatterplots of the data in each class for both polarizations are illustrated in Figure 6.2. The fraction of the training data shown is different for HH and HV, but in both cases we get an impression of how the data is distributed in the incidence angle range. The linear decaying dependency is particularly clear for the first, fifth, sixth, and seventh classes.

Table 6.1: Enumerated training data classes

Ice type	Class no.
Leads with Water/ Newly Formed Ice	1
Thick Ice, Flat	2
Thick Ice, Ridged	3
Thin Ice	4
Brash/Pancake Ice	5
Calm Open Water	6
Windy Open Water	7

6.2.1 Open water

The first dataset comprised six classes. The variance of the “open water” class was too large, and was therefore split into the subcategories “calm” and “windy”, giving a total of seven classes in our dataset.

A too large variation within a class could make the distribution of the class to be covering multiple other classes, making further classification difficulties. The mean class value in the joint open water class would also be biased, favouring the subcategory more present.

The backscatter from open water is dependent on current wind speed. With no wind, the water surface acts as a mirror, reflecting the radar signals away from the transmitting sensor. Wind gives rise to a rougher surface, which reflects the radar signals more evenly, giving more backscatter. The windy water appear brighter than the calm water in the SAR image, and the two have different incident angle dependencies. The backscatter from windy open water tends to be somewhat equal to some ice classes. Therefore, windy open water easily could be classified to one of the most similar ice classes.

The wind effect applies on water, but the hard ice surface remains the same. Ice covered with surface melt water could get a similar effect, e.g. in the melt, freeze and summer season.

One must ensure that all classes have representative data. If some range of the variation is not in the training data, this area is more likely to be misclassified to other classes (see the discussion in Section 4.3 and Figure 4.4). If parts of the open water backscatter range are not represented, the lack of data causes the supervised method to be unable to recognise this backscatter as a part of the certain class.

An other possibility for classes with large variation is to make *Parzen* functions for the classes instead of Gaussians. The Parzen is more flexible to the training data, and is not restricted to a certain distribution.

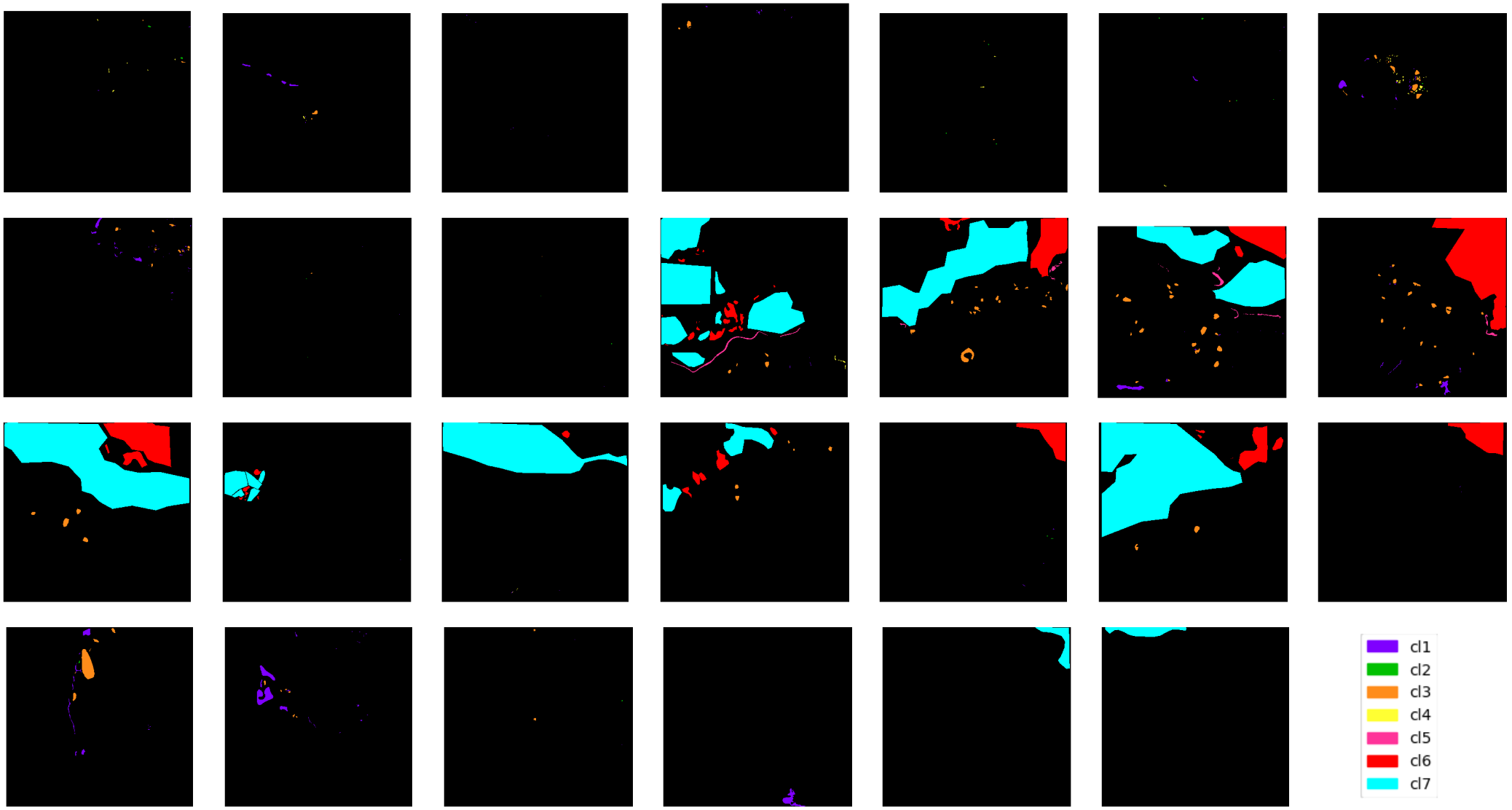


Figure 6.1: The training data polygons made in each image. The training data samples, also used for validation, are extracted from these polygons. Upper left image is scene no. 1. The images follow chronologically from left to right, line by line. The polygon class legends are in the lower right. The majority of the training pixels are from Calm Open Water (cl6) and Windy Open Water (cl7).

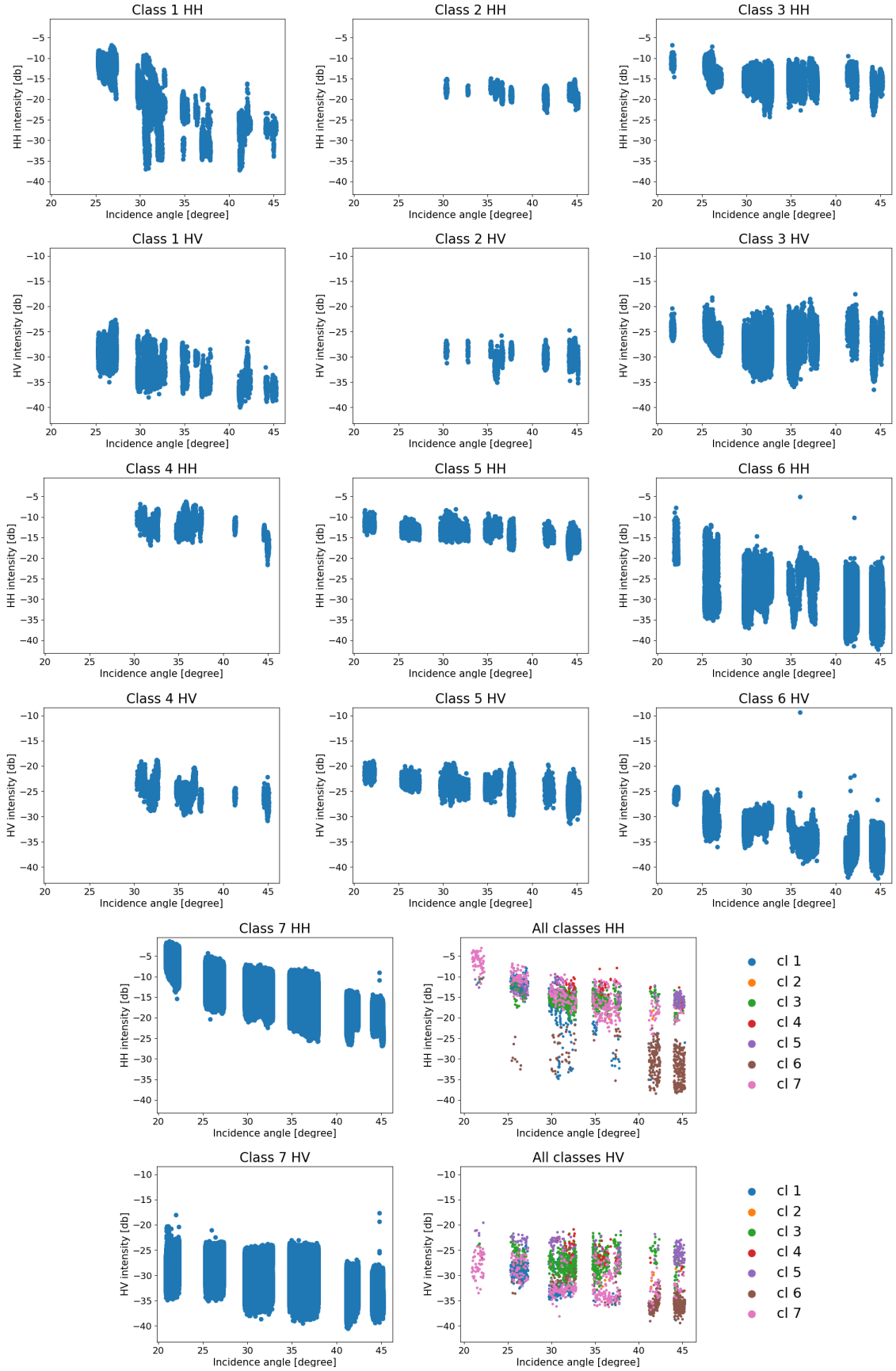


Figure 6.2: Training data in intensity-angle domain per class and polarisation (HH in row 1, 3, and 5, HV in row 2, 4, and 6). All samples plotted. Classes are listed chronologically in reading direction. All classes plotted together in the lower right, using a subsample for each class.

Part II

Methods and experiments

7

Gaussian incidence angle-dependent tubes

A Gaussian mixture is used for both the fully supervised and the segment-then-label schemes. In this chapter we describe the Gaussian incidence angle-dependent tubes, being Gaussian functions, with incidence angle-dependent means and constant variance. The shape of these structures will look tube-like. These tubes are defined in order to solve the incidence angle problem (see Section 3.6) in the classification task. To make the comparison fair between the two schemes, the models for incidence angle dependency are made equally for both methods, and adopted in both schemes.

The Gaussian tubes are determined iteratively in the segmentation, whereas for the fully supervised classification the Gaussian tubes are calculated directly from training data, as one then knows which points belong to which class.

The mean intensity is known to decay exponentially with incident angle θ, using the notation of Cristea et al. (2019):

$$\bar{I}(\theta_i) = \bar{I}_0 e^{-\frac{\theta_i}{c}}$$

where c is a surface-specific constant. This leads to a linear decay of the

log-intensities, calculated by:

$$\begin{aligned}\bar{I}_{dB}(\theta_i) &= 10log_{10}(\bar{I}(\theta_i)) \\ &= 10log_{10}(\bar{I}_0 e^{-\frac{\theta_i}{c}}) \\ &= 10log_{10}(\bar{I}_0) + 10log_{10}(e^{-\frac{\theta_i}{c}})\end{aligned}$$

using the formula for change of base in a logarithm

$$\begin{aligned}\bar{I}_{dB}(\theta_i) &= 10log_{10}(\bar{I}_0) + 10\frac{log_e(e^{-\frac{\theta_i}{c}})}{log_e(10)} \\ &= 10log_{10}(\bar{I}_0) + 10\frac{-\frac{\theta_i}{c}}{ln(10)} \\ &= 10log_{10}(\bar{I}_0) - 10\frac{1}{cln(10)}\theta_i\end{aligned}$$

This can then be written as the mean function of incidence angle for each class k with the following notation

$$\mu_k(\theta_\mathrm{i}) = a_k - b_k\theta_\mathrm{i}, \quad k = 1, ..., M, i = 1, ..., N$$

where M is the number of classes, and N is the number of samples, and

$$\begin{aligned}a_k &= 10log_{10}(\bar{I}_0) \\ b_k &= 10/cln(10)\end{aligned}$$

The intercept a_k is the log-intensity at an angle $\theta_0 = 0$ in $[dB]$, and the slope b_k is the intensity decay rate in $[dB/1°]$, both class k specific.

The Gaussian component k then have a mean vector $\bar{\mu}_k$ as a function of incidence angle, $\mu_k = a_k - b_k\theta$. The covariance Σ_k is assumed to be constant over the incidence angle range. The distribution of the mixture model with d dimensions and M components, where each component k is described by $(\bar{a}_k, \bar{b}_k, \Sigma_k)$, thus have the probability density function given by:

$$p_{X,\Theta}(\mathrm{x}, \theta) = \sum_{k=1}^{M} \frac{\pi_k}{(2\pi)^{d/2}|\Sigma|^{1/2}} e\left(-\frac{1}{2}(\mathrm{x} - (\bar{\mathrm{a}}_k - \bar{\mathrm{b}}_k\theta))^T\Sigma^{-1}(\mathrm{x} - (\bar{\mathrm{a}}_k - \bar{\mathrm{b}}_k\theta))\right) \tag{7.1}$$

where $\bar{a}_k$ and $\bar{b}_k$ are vectors with length d, and Σ us the covariance matrix. Note that for equiprobable classes, the prior π_k is equal for all classes.

8

The fully supervised classification

A Maximum Likelihood classifier is implemented, as a fully supervised method, in order to compare the efficiency of the segment-then-label method against it. The classifier is self-implemented in the Python language, and the analyses performed on this implemented version. Here follows the concept of Bayesian decision theory for the implementation, followed by some implementation details and general results using this method.

8.1 Bayesian decision theory

Let's say that $p(x|\omega_i)$ is the conditional distribution of the data given affiliation to class ω_i, and $P(\omega_i)$ is the prior probability for class ω_i. Using Bayes rule, see Appendix A, the conditional probability of a class given some data points x, also called the posterior probability is:

$$P(\omega_i|\mathrm{x}) = \frac{p(\mathrm{x}|\omega_i)P(\omega_i)}{P(\mathrm{x})}$$

where $P(\mathrm{x})$ is the probability density function for the data points x.

Bayesian decision theory is based on determining the most likely possibility.

According to Bayes classification rule in a classification task with M classes, $\omega_1, \omega_2, ..., \omega_M$, data points x are classified to class ω_i if the probability for the point being in class ω_i is higher than the probability of belonging to other classes:

$$\mathrm{x} \in \omega_i \quad if \quad P(\omega_i|\mathrm{x}) > P(\omega_j|\mathrm{x}) \quad \forall j \neq i$$

The decision boundary x_0 that optimally separates the classes $\omega_i, i = 1, ..., M$, is found by the discriminant functions $g_i(x), i = 1, ..., M$:

$$g_i(\mathrm{x}) = p(\omega_i|\mathrm{x})$$

using Bayes rule, where $p(\omega_i|X) \propto p(X|\omega_i)P(\omega_i)$

$$g_i(\mathrm{x}) = P(\omega_i)p(\mathrm{x}|\omega_i)$$

such that the decision boundary is drawn where

$$x_0 : \quad P(\omega_i)p(\mathrm{x}|\omega_i) = P(\omega_j)p(\mathrm{x}|\omega_j)$$

This is called the Maximum A Posteriori (MAP) approach, as it maximizes the posterior as the discriminant function.

8.1.1 The Maximum Likelihood classifier

Equal priors for all classes i result in the Maximum Likelihood (ML) approach, where the discriminant function is simplified to be a maximization problem of the likelihood functions for each class:

$$g_i(X) = p(X|\omega_i)$$

such that the decision boundary is drawn where

$$x_0 : p(\mathrm{x}|\omega_i) = p(\mathrm{x}|\omega_j)$$

8.2 The Mixture of Gaussian componets

For the supervised ML classifier the class membership z_{ik} is 1 only for samples x_i with the class label of class ω_k:

$$z_{ik} = \begin{cases} 1, & \text{if } x_i \in \omega_k \\ 0, & \text{if } x_i \notin \omega_k \end{cases}$$

as the labelled training data belong exclusively to one class.

In a supervised task, where labels are provided for the training data, the mixture components will be deduced from the samples with specific labels. The expressions for the parameter equations for the mean hyperparameters and the covariance parameter follows for the mixture components (see Chapter 7) are found with the following equations, derived from Equation 7.1:

$$\mathrm{a}_k = \frac{\sum_{i=1}^{n_k} \mathrm{x}_{i|x\in\omega_k} + \mathrm{b}_k \sum_{i=1}^{n_k} \theta_i}{n_k}$$

$$\mathrm{b}_k = \frac{-\sum_{i=1}^{n_k} \theta_i \mathrm{x}_{i|x\in\omega_k} + \mathrm{a}_k \sum_{i=1}^{n_k} z_{ik}\theta_i}{\sum_{i=1}^{n_k} \theta_i^2}$$

$$\Sigma_k = \frac{\sum_{i=1}^{n_k} (\mathrm{x}_{i|x\in\omega_k} - (\mathrm{a}_k - \mathrm{b}_k\theta_{i|\theta\in\omega_k}))(\mathrm{x}_{i|x\in\omega_k} - (\mathrm{a}_k - \mathrm{b}_k\theta_{i|\theta\in\omega_k}))^T}{n_k}$$

The hyperparameters a and b are dependent on each other, and are found via a linear regression of the training data input in the joint space of log-intensity and incidence angle. As the covariance is assumed to be constant with incident angle, the covariance is calculated for $\theta_0 = 0$, such that the expression is reformulated:

$$\Sigma_k = \frac{\sum_{i=1}^{n_k} (\mathrm{y}_{i|x\in\omega_k} - (\mathrm{a}_k))(\mathrm{y}_{i|x\in\omega_k} - (\mathrm{a}_k))^T}{n_k} \tag{8.1}$$

where $\mathrm{y}_i = \mathrm{x}_i - \mathrm{b}_k\theta_i$ are the sample points projected to the $\theta = 0°$ and a_k still is the mean value for all samples projected to $\theta = 0°$ incidence angle.

8.3 Implementation

The Bayesian ML, based on equiprobable normally distributed classes, is implemented. The features used are the cross- and co-pol channels (HV and HH), along with the incidence angle. The classes are considered Gaussian tubes (see Chapter 7), meaning the classes are components of a Gaussian mixture. The classification scheme is as follows:

1. The training stage consist of estimating the parameters of the Gaussian distribution for each class. The mean hyperparameters a and b

   $$\mu_k(\theta_i) = a - b\theta_i$$

   are found by performing a linear regression in the intensity-incidence angle space. The covariance of the samples in class k is found by the $\theta = 0$ projected sample points, as in Equation 8.1.

2. A linear discriminant function is made for each class. The monotonic logarithm function is used on the discriminant function to get

$$g_k(\mathrm{x}) = ln(p(w_i|\mathrm{x})) \qquad w_i : \mathrm{class}i$$

applying Bayes rule

$$= ln(p(\mathrm{x}|w_i)P(w_i))$$
$$= ln(p(\mathrm{x}|w_i)) + ln(P(w_i))$$

Inserting a gaussian distribution and excluding priors, as classes are equiprobable, and terms that are equal for all classes.

$$g_k(\mathrm{x}) = -\frac{1}{2}(\mathrm{x} - \mu_k)^T \Sigma_k^{-1}(\mathrm{x} - \mu_k)$$

A multivariate normal function is used for implementing this discriminant function.

3. The class k for which $g_k(x)$ is maximized for a data point x_i, is set as class for this data point.

8.3.1 Training data

Training data for the ML classifier are extracted from the manually drawn polygons, and contains the HH, HV and incidence angle features (see Chapter 6). The trained ML classifier is able to recognize seven classes, each described by a slope, an intercept, and a covariance matrix.

8.3.2 Decision

The trained classifier is then utilized for classification of each pixel. As each class k is described by the set (a_k, b_k, Σ_k) for $k = 1, ..., M$, each class will have a multivariate normal probability density function as given in Equation 7.1. The samples to classify are given as $(\mathrm{x}_i, \theta_i) \quad \text{for} \quad i = 1, ..., N$, where N is the total number of samples. (x_i, θ_i) are inserted into the M multivariate normal functions, one function for each class. The decision for one pixel is then the class associated with the highest probability for that pixel. This is a one point sample against a set representative measure, where the set is represented with both mean and variance (see Section 2.1.1).

A programming efficiency issue occurs if expansion of the classification to the whole image is done by a for-loop over all the pixels. As this takes rather long time, also when the image is downsampled, the decision is optimized by looping over the classes instead, as described in the preceding section.

8.4 Validation

A 10-fold cross-validation is performed for validation of this method separately. The training data set is split into 10 equally sized parts. For each of the 10 runs, one of the parts functions as validation, whereas the nine parts left are used for training the classifier. The procedure of training and classification on the validation set is run 10 times, and the classification accuracy for all runs are averaged to get the final classification accuracy.

The training data merged from the original-sized 27 images are stratified such that each class has a training sample size of 100 000 samples. The 10-fold cross-validation then produces an accuracy of 56%. A 100-fold cross-validation gives an accuracy of $56.24 \pm 0.06\%$.

Using the whole training data set without stratification yields an overall cross-validated accuracy of 60%. As this data is dominated by open water, we conclude that the classification of the open water samples pulls the accuracy upwards.

8.4.1 Examples of the visual results

For visual inspection of the result, an example of image classification is shown in Figure 9.1f. Land is masked out, and we will only discuss the image parts which are not masked. The left and uppermost area represents open water. The classified image shows that a large part of the open water area is misclassified as ice. Nevertheless, edges, and the main structures of the image are captured. Even though the first swath is significantly brighter than the rest, and therefore often is misclassified, we see that using the incidence angle dependency corrects for this.

Another example is shown in figure 9.2f for an image of an area consisting mostly of ice. This is a visibly good result, as the thin, long areas of thin ice are recognized, as well as the large areas of thick ice. Also in this example we see that there is no clear class deviation from near-range to far-range.

9

Segmentation and labelling

An automatic segmentation algorithm, which will be further described in this chapter, is provided by A. Doulgeris. The code for this algorithm is used in its Python version. Even though the segmentation algorithm is ready-implemented, the preprocessing before the segmentation and the framework for the testing are implemented specifically for this thesis. All analyses, the testing, and the validation are implemented in Python.

9.1 Segmentation

We use a segmentation algorithm (hereafter: "the segmentation") developed for multi-polarization SAR images, that is an automatic Expectation-Maximization (EM) algorithm utilizing a Gaussian Mixture Model (GMM). It considers the incidence angle, and optionally does a MRF smoothing. The determined clusters are Gaussian tubes, i.e. Gaussian mixture components, having incident angle dependent means (see Chapter 7), with specific incidence angle dependency for each component. The obtained segment information is used for expanding the segmentation to the whole image. In the end an optional MRF smoothing does a contextual smoothing in order to achieve a less noisy result. This method is efficient enough for operational use as it is fast and determines the number of classes automatically (Cristea et al., 2019; Doulgeris and Cristea, 2018; Cristea and Doulgeris, 2018).

This method is an extension of the Extended Polarimetric Feature Space (EPFS) method presented in Doulgeris (2013). The EPFS method extracts features from the local neighbourhoods, and the features are transformed to be Gaussian or Gaussian-like, to make them suitable for a Mixture of Gaussian clustering. This method extends the basic Mixture of Gaussian method by automatically determining the number of classes, and by handling of the incidence angle dependence within the classes. The incidence angle awareness is useful for wide incidence angle range images, whereas the EPFS method still works fine for narrow-swath images, where the incidence angle decay is negligible.

9.1.1 Features

The EPFS method contain polarimetric and textural features, and our segmentation is also capable of taking such additional features, along with multiple polarization layers. We are only using the log-intensity layers for the HH and the HV channels. The intensity layers must be in the log-intensity versions given in decibels, as the process is based on the linear decay of log-intensities. The use of other features than the intensity layers is out of the scope for this thesis.

9.1.2 The Expectation-Maximization algorithm

As the samples in a clustering task do not have labels, the mixture components must be found iteratively. The lack of labels causes the idea of an *incomplete dataset* (Theodoridis and Koutroumbas, 2009, p.45).

The EM algorithm is a probabilistic method, solving tasks with incomplete datasets. E.g. for problems involving mixture models where labels are not present. It is a cost function optimization-based clustering algorithm, the cost function to iteratively optimize being the complete likelihood of both observed and latent data in the incomplete dataset:

$$Q(\Theta) = E\{LL(\Theta\}$$

such that the optimization is given as

$$\frac{\partial E\{LL(\Theta)\}_{Q(Z)}}{\partial \Theta_i}$$

The latent data, which could be the labels or class membership function, gives the opportunity to perform the statistical expectation. The expectation step updates the expectation of the latent data, based on the current iteration

parameter estimates. The parameters are then updated in the maximisation step, by maximising the complete likelihood of the parameters.

The algorithm runs iteratively over the two steps until convergence is reached for the parameters, and each component is associated with the optimal mean and covariance parameters. The latent data is the membership function $z_k, k = 1, \ldots, M$, where M is the total number of components. The expectation step thus updates the membership weights z_{ik}, such that it denotes the expected probability that pixel i belongs to the class k. A derivation of this expression is shown in Appendix B.

$$E\left\{z_{im}\right\}_{p(Z|X;\Theta)} = \frac{p(x_i|z_{im}=1,\Theta)}{\sum_{k=1}^{M} p(x_i|z_{ik}=1,\Theta)}$$

For GMM a Gaussian distribution is inserted for $p(x_i|z_{im}=1,\Theta)$.

The maximization step updates the parameters $\Theta = [\mu(a_k, b_k), \Sigma_k]$. The mean function holds 2 parameters, a and b (see Chapter 7). The parameter update expressions are shown here, and the derivations can be found in Appendix B.

$$\begin{aligned}
\mathrm{a}_k &= \frac{\sum_{i=1}^{n} z_{ik}\mathrm{x}_i + \mathrm{b}_k \sum_{i=1}^{n} z_{ik}\theta_i}{\sum_{i=1}^{n} z_{ik}} \\
\mathrm{b}_k &= \frac{-\sum_{i=1}^{n} z_{ik}\theta_i \mathrm{x}_i + \mathrm{a}_k \sum_{i=1}^{n} z_{ik}\theta_i}{\sum_{i=1}^{n} z_{ik}\theta_i^2} \\
\Sigma_k &= \frac{\sum_{i=1}^{n} z_{ik}(\mathrm{x}_i - (\mathrm{a}_k - \mathrm{b}_k\theta_i))(\mathrm{x}_i - (\mathrm{a}_k - \mathrm{b}_k\theta_i))^T}{\sum_{i=1}^{n} z_{ik}}
\end{aligned}$$

9.1.3 Goodness-of-fit testing

The automation of the EM algorithm is made by running this procedure for a range of numbers of classes. The model with the best fitting amount of classes is chosen when the goodness-of-fit criterion is first met. In principle this criterion is a Pearson's χ^2-test, comparing the experimental values with theoretical values from the χ^2 distribution. Doulgeris (2015, section II.C).

9.1.4 Cluster decision

A clustering could be performed based on the resulting probability for a sample belonging to a class, given the mixture components achieved from the converged EM scheme. The final clustering is then a hard Bayesian decision,

where a sample belongs to the most probable class. The decision could also be stochastic based on the probability for the different classes. The first case is used in this segmentation.

9.1.5 Markov Random Field Smoothing

The problem discussed in Section 4.2, about the possibility of clustering a point to the wrong distribution due to overlapping distributions, is solved by applying a MRF smoothing. The smoothing is optionally done after the segmentation from the EM algorithm. It is performed for a visibly better result.

For images, the MRF fields have the trait that a pixel directly depends only on the other pixels in its local neighbourhood of a specified size. A pixel is not directly independent, but rather indirectly dependent through the Markov property, on all other pixels in the image (Elachi and Van Zyl, 2006, p.434). The Markov field does an adjustment of the class priors based on the local neighbourhood, that leads the Bayes classifier to possibly reclassify pixels in a probabilistically rigorous manner.

For a fair comparison of the method, when no MRF smoothing is implemented for the fully supervised method, the MRF smoothing is also not used after the segmentation.

9.1.6 Tuning possibilities

The segmentation has two possible tuning parameters. These are the number of looks in the input feature and a subsampling option.

Number of looks

The GRDM product is already multilooked as a part of the product's nature. The segmentation algorithm does not do an additional multilook, but the user provides the algorithm with inputs that are either additionally multilooked or not. The benefits of using additional looks are discussed in Section 5.4.

The segmentation is initially run to see how the multilook tuning plays a role in the algorithm. Different number of looks are used in separate runs: filtersizes of 15x15, 9x9, 5x5 and 3x3 pixels. A 15x15 filter gives a resolution of 1 395 m x 1 305 m, thus an ice area needs to be at least of this size to be distinguishable from other ice areas. After using a 5x5 px filter, this size is 465 m x 435 m. Two different ice types will smooth over in each other if they are nearer than the

resolution distance. This means that one for example may loose narrow leads. For larger multilook filters the thermal noise will be too broad, disturbing a larger part of the image.

We consider a 5x5 pixels multilook filter as appropriate, as the speckle and noise is sufficiently smoothed, but the small and tiny ice areas are still visible.

Subsampling and sensitivity

A sub-sampling option is set before the clustering, for controlling the number of samples used for the training. This option restricts the amount of pixels that the segmentation training uses for tuning the mixture components. The clustering process speeds up and gives a quicker result by using fewer samples. This comes at the cost of the clustering sensitivity, which increases with sample size. A larger sample size give higher sensitivity, and a smaller sample size less sensitivity.

By sensitivity is here meant the variance of the Gaussian curves. The lower the variance, the finer the Gaussian curves, and the easier it will be to distinguish between partly overlapping classes. The histogram used for the Pearson's χ^2-test will be smoother with more samples. The goodness-of-fit test will thus achieve its threshold more rapidly for less samples, and stop at a stage with fewer clusters. With high sensitivity, the variance is lower, thus it is easier to distinguish more clusters. With low sensitivity the algorithm gets a challenge by distinguishing overlapping classes, whereas this could be a benefit if fewer classes is preferred.

Different levels of sub-sampling are tried for a variation of multilook levels. We want around 3-4 classes more than there are in the image, to have the opportunity to label more segments with the same label. With 5x5 multilook a subsampling of 80 000 samples is chosen, as this gives an appropriate amount of segments in a realistic amount of time.

9.2 Labelling

From the literature 1.2, it is suggested to use a distance measure. Moen et al. (2015) found that the Mahalanobis distance outperformed the four other distance measures used for labelling of the Gaussian-like segments. Based on this our labelling uses the Mahalanobis distance

$$d(\mathrm{x}, \mathrm{y}) = \sqrt{(\mathrm{x} - \mathrm{y})^T \Sigma^{-1} (\mathrm{x} - \mathrm{y})}$$

The Mahalanobis distance is calculated from two normal functions; the Gaussian functions resulting from the segment part is compared against the Gaussian functions obtained from the training data. The Mahalanobis distance is related to the likelihood of the Gaussian function. The logarithmic version of the Gaussian likelihood is given by

$$LL(\Theta) = -\frac{d}{2}log(2\pi) - \frac{1}{2}log(|\Sigma|) - \frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)$$

where d is the number of dimensions, and μ and Σ are the Gaussian parameters. Excluding all constants, this is related to the Mahalanobis distance, but with an additional covariance term. As the Mahalanobis distance is found to be well performing for labelling, we assume that using the likelihoods will give the same level of performance as shown for the Mahalanobis distance.

Two approaches of using the likelihoods in the labelling are considered. The first is the *total likelihood*. The likelihoods for belonging to each segment are calculated for all training data points, based on the segments' slopes and the training point's location. Then each segment is labelled with the training data class' label with the highest normalized total likelihood.

The second approach is a *majority counting* among the likelihoods for a segment's pixels. The pixels within a segment are compared to the training data class slopes. Each segment pixel gets a temporary class label, being the most likely training data class, before the labelling of the whole segment is set as the majority class among its own pixels.

The majority counting may be a main mechanism to add benefit for the segment-then-label method with respect to contamination. This is because the majority vote will essentially filter outliers. The total likelihood approach would still suffer from the contamination problem, as it does not have the outlier filtering, but blends in all likelihoods in the total likelihood.

9.2.1 Important consideration

The training data will contain seven classes, as we have training for those. The results from the segmentation may on the other hand contain different amounts of classes, depending on the time and area of the acquisition and the parameter tuning used. Some images simply do not contain certain ice types. Therefore, some of the classes represented in the training data may not appear as labels when classifying. It may also be that more than one cluster from the segmentation happen to have the same class as the one with the shortest distance, resulting in many clusters having the same label.

9.3 Segmentation and labelling examples

Visual results from the segmentation on two different images are shown in Figures 9.1 and 9.2. The HH and HV log-intensity images processed with a 5x5 multilook are shown for the respective scenes in Figures 9.1a and 9.1b, and Figures 9.2a and 9.2b. These are used as references for comparing the classification results against the brightness images. The masked-out areas (black) are not considered in the analyses, as including noise and land that do not follow the model would affect the slopes too much. Figures 9.1c and 9.2c show the segmentation results for both images, using a subsample of 80 000 and processed with a MRF smoothing. The corresponding slopes for each segment, together with the scatter for each segment, are shown in both polarizations in the Figures 9.1d and 9.2d. The results after labelling the segments are shown in Figures 9.1e and 9.2c for the respective scenes. The MRF smoothing is applied here as an example of the clear visual improvements it adds. The difference is clear comparing with the ML result in Figures 9.1f and 9.2f, where MRF smoothing is not applied. Note that in the later comparison we do not use the MRF smoothing at all.

In Figure 9.1c the segmentation determines 15 segments. The edge between ice and water is found. We see that the edge consists of at least two ice segments: the bright green and the orange. As the open water area contains a wide range of brightness values even for a single incident angle, it contains many segments. The inner ice area also consists of many segments. This seems to be somewhat range dependent, as the midswaths consist of some particular segments, whereas the far-range-swaths consist of other segments. Different clusters are made for the ice region and the open water region. The segments on the open water and on the ice do not seem to be shared. The amount of the same segments on ice and water is marginal. Therefore, the segmentation seems to be working well for this image. Remember that after the labelling, the many segments are merged into the fewer classes.

The segmentation result in Figure 9.2c is an area over ice, and contains a small area of open water in the middle of the first swath (left in image). The segmentation is dominated by the dark blue and dark green segments. The long, thin areas across the image are four different segments. The open water has bright green and orange segments. Thus, this image seems to be realistically segmented, according to what is seen in the brightness images.

Notice that even though we have seven training classes, the segmentations have achieved up to 15 segments. For the other images the number of segments is from 4 to 14. The number of segments is deliberately higher than the number of classes, as more segments are allowed to belong to the same class.

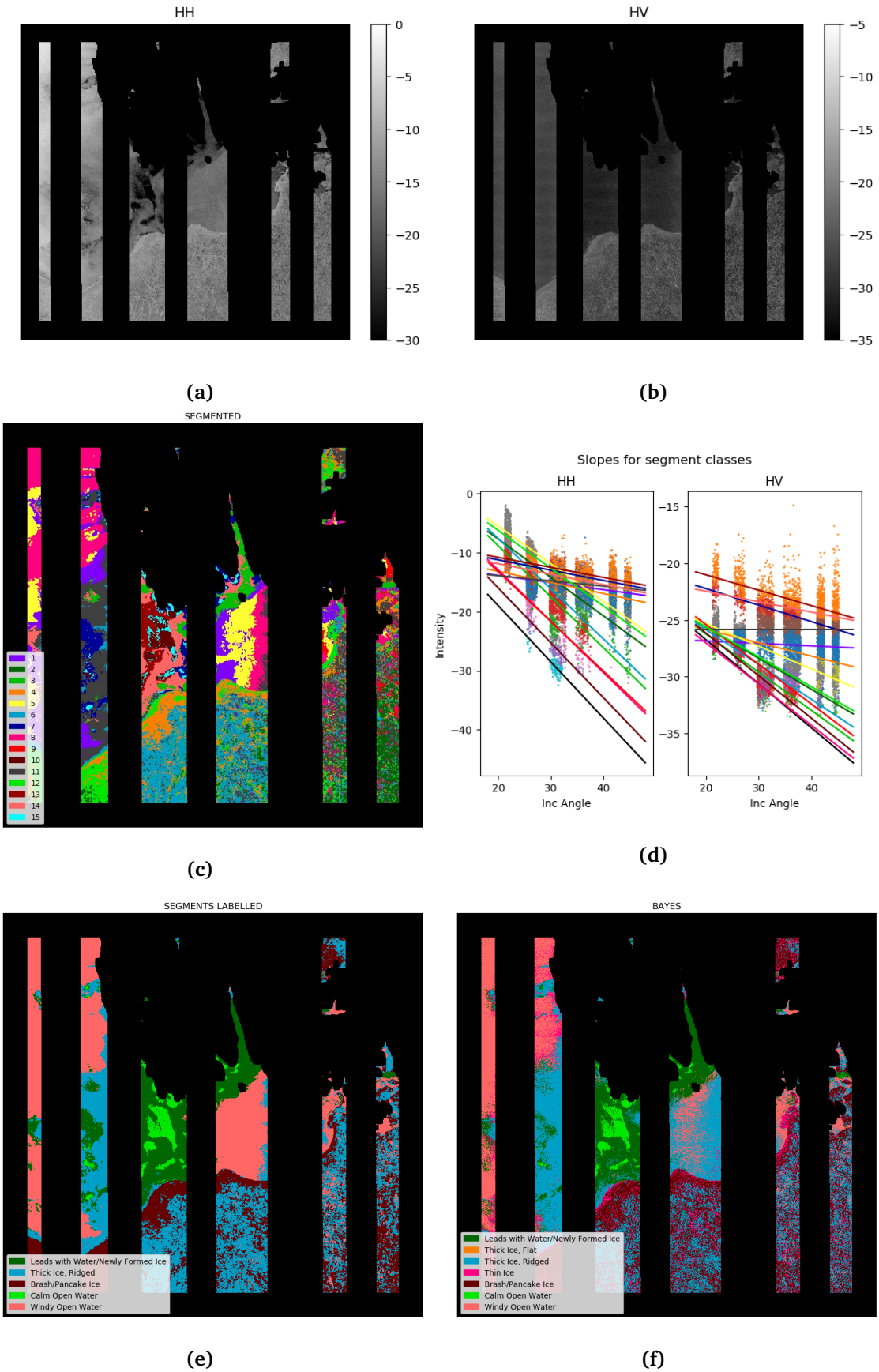
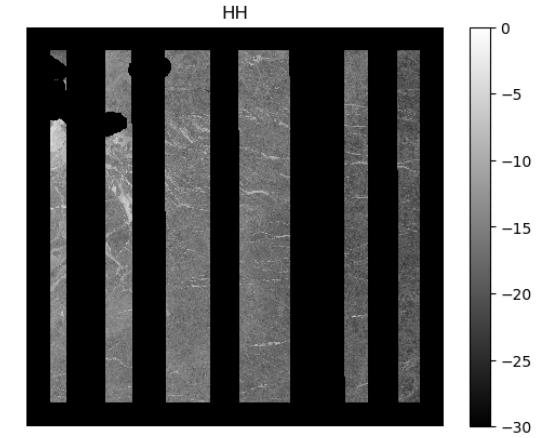
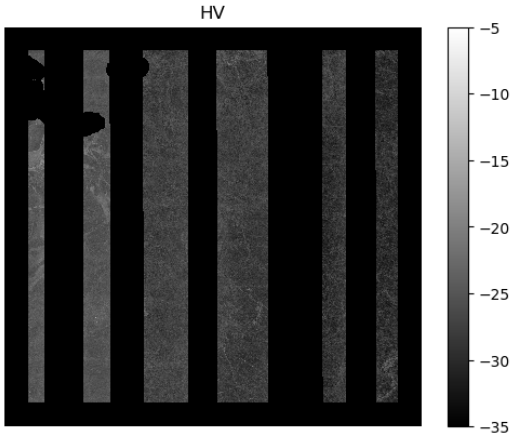


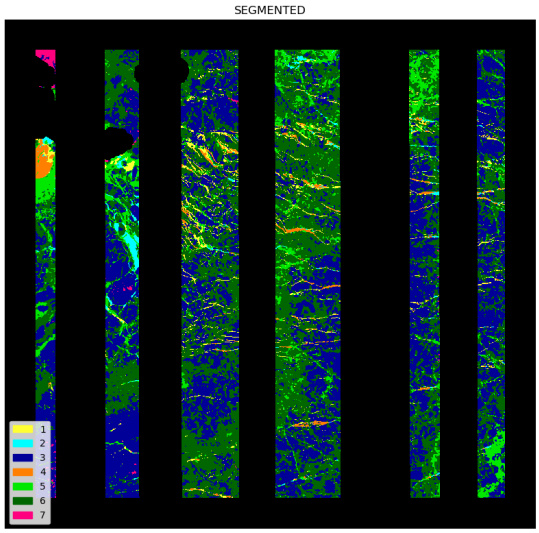
Figure 9.1: Image no. 11, with areas of ice, open water, and land. (a) The HH and (b) the HV log-intensity images. (c) Segmentation results. (d) The segment slopes in both polarizations. Colours do not match segments. (e) Result using the segment-then-label method. (f) Result using the ML classifier.



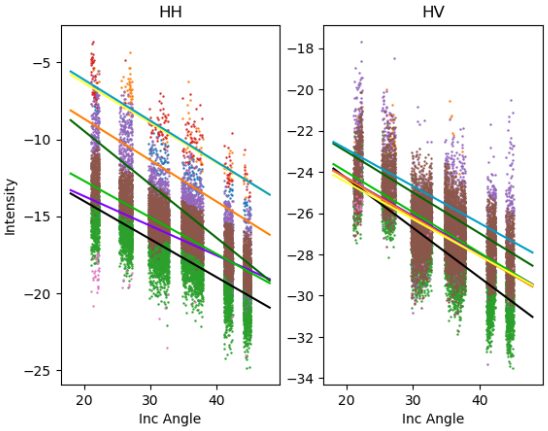
(a)



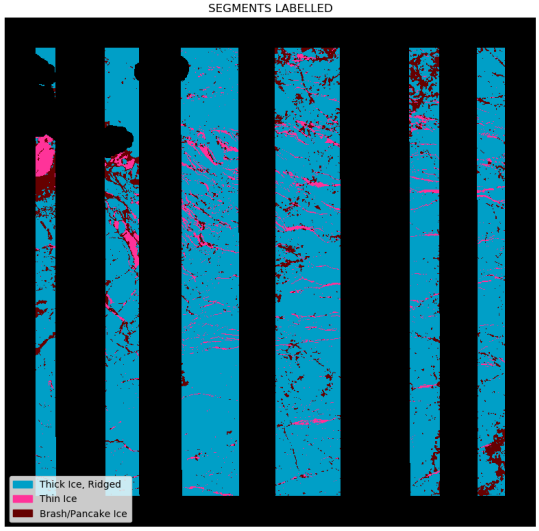
(b)



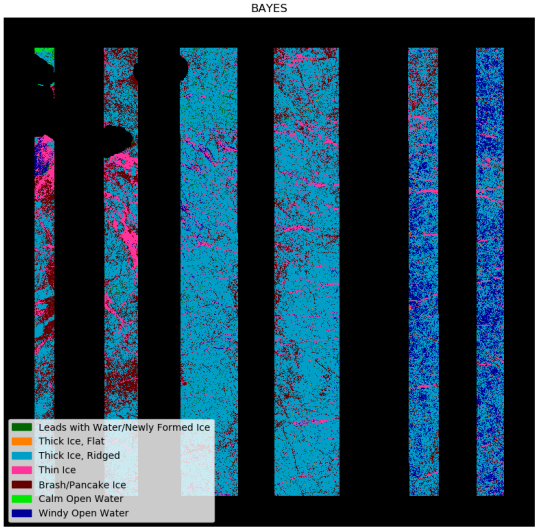
(c)



(d)



(e)



(f)

Figure 9.2: Image no. 5, for the most containing ice. (a) The HH and (b) the HV log-intensity images. (c) Segmentation results. (d) The segment slopes in both polarizations. Colours do not match segments. (e) Result using the segment-then-label method. (f) Result using the ML classifier.

10

Comparison of the two methods

This chapter focuses on the comparison of the methods' performances for two cases: (1) The amount of training data and (2) the contamination or mislabelling in training data.

Two cases

The training data amount case (1) is chosen, as the background for unsupervised learning is to not use training data. One wants to add as little training data as possible in the labelling stage, to get the same good results as for a direct supervised method. The supervised methods have shown good performance when enough training data is available, but we want to test if the segment-then-label method can outperform it for a small training data amount. The contamination case (2) is chosen as the second test case, as it is important to know how the two methods behave with certain amounts of outliers in the training data.

The maximum amount of training data is when using all training data samples. A training data amount of "1 000" means that 1 000 samples are randomly picked for each class to train the classifier. Note that the whole downsampled data set in first split into training (80%) and validation (20%), and then the

80% training data is restricted. Thus, the size of the validation sample remains the same size when training data size is restricted.

The contamination test is done using 1 000 samples per class. A "10%" contamination means that 10% of the samples within each class is randomly mislabelled to the other classes. In such a way some of the training data samples interchange labels, one class getting labels of other classes.

Performance measure

Classification accuracy is used as the measure of the classification performances. Two types of classification accuracies are tested. The first is the *total accuracy*, being the fraction of all correct classified data points, not regarding the classes. The second is the *mean class accuracy*, calculating the classification accuracy for each class separately, before averaging over the classes. To avoid any class dominance when using the total accuracy, one has to ensure that the amount of validation samples is the same for all classes. The mean class accuracy bypasses this problem by letting each class have the same influence, regardless of the number of samples per class.

The accuracy measure needs to be carefully implemented, as there are many different images, containing a various amount of validation samples from the different classes. When a few classes dominate in an image, the accuracy will be heavily influenced by these classes.

Independent training and validation data

The classified pixels for evaluating the method should be independent of the pixels used for training the classifier. If the results are checked against the exact same data as they were trained for, the accuracy naturally will be high, as the classifier is trained to classify exactly these points.

The data from the polygons of the reduced size image is split up to training (80%) and validation (20%) sets. This is done for each image, such that there for all images are 20% of the polygon-points that are used for the validation and 80% for training, not regarding classes. The 80% training data from each image is collected and joined to train the ML classifier and the labeller.

As there is a limited amount of training data in each image, all these images are used for calculating the accuracy measure. The 27 scenes are run through both methods, such that each scene has two resulting images, one for the supervised method and one for the segment-then-label method. Note that each image has

a limited amount of training data (see Figure 6.1 and Table C.2). From each image is extracted the number of pixels correctly classified, along with the total number of training samples in the image. The validation is joined for all images, such that a validation is performed on the total amount of validation points.

The numbers are joined such that the total accuracy is computed as:

$$\frac{\#\text{ correct classified pixels from all images}}{\#\text{ validation pixels in total from all images}}$$

The mean class accuracy is computed as:

$$\frac{1}{\#\text{ classes}} \sum_{i}^{\#\text{ classes}} \frac{\#\text{ correct classified pixels from all images, for class i}}{\#\text{ validation pixels in total from all images, for class i}}$$

Repeated runs

The classifications are run 100 times to get valid results, including both the mean of hundreds and the variation of hundreds for the accuracy. In the graphs, the error-bars are plotted as 95% confidence intervals using the student-T distribution.

10.1 Graphical results

Figure 10.1 shows the total accuracies measured for varying percentage of contamination, for both methods. Figure 10.2 shows the mean accuracies for the same case, and Figure 10.3 the mean accuracies for restricting the sizes.

A tendency is that more contamination in the data makes more pixels to be classified as open water. The large sample size of open water makes the total accuracy curve to increase for more contamination, as seen in Figure 10.1. The reason for the open water to be well classified, is that there may be more likely for the ice classes to intermix slopes, than to mix their slope with open water. The open water has a more distinct distribution, more different from the ice classes. As the data contains a large amount of open water samples, a safer measure is the mean class accuracy.

The mean accuracy graphs decrease by both fewer samples (Figure 10.3) and more contamination (Figure 10.2). The decrease with contamination is small, but the decrease after 60% is larger for the fully supervised than for the segment-then-label.

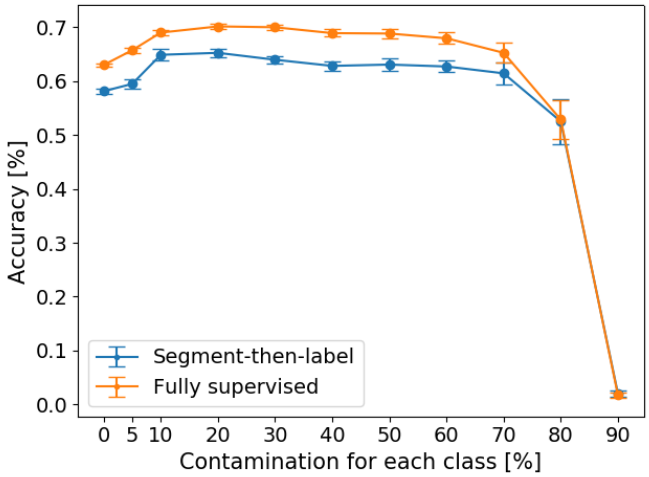


Figure 10.1: Total accuracies measured for varying percentage of contamination in each class, and using a training (80%) - validation (20%) split to the downsampled training data set. Each point is a mean of hundred independent runs, and the errorbars show the 95% confidence intervals. Due to a larger amount of open water among the validation points, which happen to be better classified by more contamination, we see that the total accuracies are increasing for both methods. Both curves start to decay at around 70% contamination in each class, which is not quite reasonable.

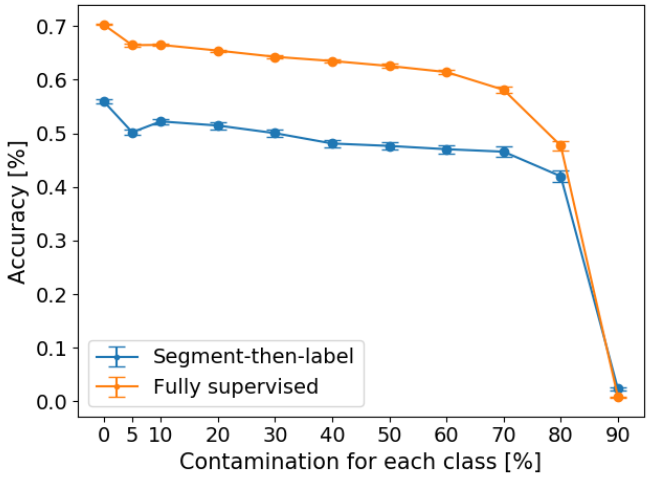


Figure 10.2: The mean class accuracies for varying percentage of contamination in each class. A training (80%) - validation (20%) split is done to the downsampled training data set, and the accuracy is calculate from the validation data. Each point is a mean of hundred independent runs, and the errorbars show the 95% confidence intervals. The fully supervised method (orange) overall has a better classification accuracy than the segment-then-label method (blue). The decay rate is approximately equal for both methods.

The accuracy of the fully supervised method (in orange) in general becomes higher than for the segment-then-label method (in blue). Looking at the class accuracies, we notice that the "leads/newly-formed ice" class makes the mean class accuracy for segment-then-label to be lower. The method simply classifies this class with a low accuracy. Calculating new mean class accuracies, leaving out the leads class, the accuracy of the segment-then-label method is still lower.

10.1.1 The importance of both visual and statistical results

Ideally, statistical measures of the method performances should be compared against each other. Subsequently, decisions are best made based on statistical significance. In this work we classify a number of images, and the validation data are spread on all images. Only a tiny part of all classified pixels are validated. Thus, the accuracy measure cannot describe the classification of the whole image, but indicates how some small parts of the images are classified.

In the ideal case, a whole image is labelled as a reference, and used as validation

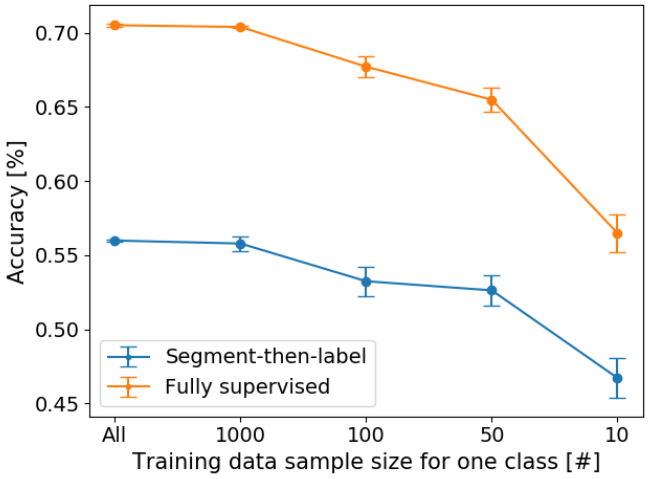


Figure 10.3: The mean class accuracies for varying number of training sample for each class. The downsampled training data is split into training (80%) and validation (20%). Each accuracy point is a mean of hundred independent runs, and the errorbars show the 95% confidence intervals. The fully supervised method (orange) overall has a better classification accuracy than the segment-then-label method (blue). The decay rate is similar down to a 100 training samples per class. For less than 100 samples per class, the decay is steeper for the fully supervised (orange) than for the segment-then-label method.

for this image's accuracy. As this is not our case, visual inspection of the images is helpful, as it will give an impression of how the classifications carry out image-wise.

10.2 Visual comparison results

The classification results are illustrated visually in example 1 Figures 10.5/10.6, example 2 Figures 10.7/10.8, and example 3 Figures 10.9/10.10 on the next pages. These images are chosen to illustrate the results, as they contain a diversity of ice classes and open water areas. A description of how to read the pages is presented as two "pages" in Figure 10.4.

The first page in an example starts with a row containing the HH and HV log-intensity images for both polarization channels. For the sake of visual impression, the HH and HV log-intensity images are clipped to the log-intensity range shown in the colorbars. The second row shows the "perfect" classification for both methods using the training data from the original-sized image. The

Figure 10.4: Description on how to read the figure pages with results.

Example nr X, Page 1/2		
	C1	C2
R1	HH log-intensity	HV log-intensity
R2	Fully supervised: The "perfect"-classified	Segment-then-label: The "perfect"-classified
R3	Fully supervised: The "best" classified	Segment-then-label: The "best" classified

Example nr X, Page 2/2		
	C1	C2
R1	Fully supervised: 50 samples per class	Segment-then-label: 50 samples per class
R2	Fully supervised: 10 samples per class	Segment-then-label: 10 samples per class
R3	Fully supervised: 70% contamination	Segment-then-label: 70% contamination

third row shows the “best” classification, using the reduced training data from the downsampled image. Bayes ML classifier takes the first column, and the segment-then-label method takes the second.

The first row on the second page of an example is results for using 50 sample points per class for training the classifiers. The second row is for using 10 sample points per class. The third row is using 70% contamination in the training data. Notice that the same samples and division are used for producing both the ML and the segment-then-label results in a certain row.

The “best” and “perfect” classification results

The “perfect”-classified images are made using the large training data set from the original-sized images. The “best” results are made using the reduced training data set from the downsampled images. The classes of this training data has different sample sizes, ranging from 1 600 to 2 000 000 per class (Table C.2). The classes with more training data could be more well-trained, but the 1 600 samples should still be sufficient to train a good classifier, especially taking into account the large range of angles.

The difference in the “perfect” and the “best” images, can be considered in Figure 10.7. The “perfect”-classified (second row) shows somewhat similar results for both methods, with large areas of light blue ridged thick ice, and dark green areas of leads/newly-formed ice around the islands. The fully supervised method seems to misclassify the area in the lower right corner to open water. Further, the methods do not agree about the small areas of thin and brash ice (red and pink). The “perfect” images are fairly equal and seem reasonable. In the “best” result (third row) the leads area is changed into calm

open water and flat thick ice. The ML even classify the rightmost part of the image to flat thick ice, which previously was ridged. We also notice the area between the island (upper left), that is classified to flat thick ice or calm open water when using the reduced size training data, but as leads/newly-formed ice, using the original training data size.

Considering the image in Figure 10.9, there is not too much difference between using the large and the downsampled training data for the segment-then-label. For the ML classifier, on the other hand, the difference is clear by a large area of ridged thick ice in the lower part of the image. The open water is partly classified to open water, partly leads and flat thick ice. For the downsampled set the open water seems to be better classified, but the thick ice areas are a lot more mixed. In Figure 10.5, the ML classifier similarly does better either on ice or water for the different sample sizes.

10.2.1 The results from the classified scenes

Using 50 and 10 samples per class (the examples' page 2, first and second row):

By using the ML classifier we see that the smaller sample size results in a classified image which is more divided in range direction (Figure 10.6 first coloumn, first and second row). The image seem to be divided swath-wise by the classes. In the example with 50 samples per class, the leftmost part of the ice is classified as leads and thin ice, the middle part as ridged thick ice, and the right part as flat thick ice. The same effect is seen when only 10 samples per class are used, but different ice types dominate due to randomness in the sampling. From left to right the ice is now divided into leads and calm open water, ridged thick ice and thin ice. This is an effect of changing training data slopes, such that the mean values, particularly for near and far-range, are significantly different. For reduced sample size in Figure 10.8 (first column, second row), the outcomes are also changed in the far range swath. Here, the thick ice and the ridged ice probably have changed their slopes relatively to each other, such that a large area which probably are ridges, are classed to flat thick ice. The similar effect is also seen in (first coloumn, first row) between ridges and thin ice in the near range, and in the example in Figure 10.10 (first column, first and second row).

For the segment-then-label, the effect is less distinct. In Figure 10.6 (second column) there is, though, also mixing between flat and ridged ice. For the 10 samples case (second row), almost the whole ice area, as well as one of the major segments on the water, are classified as ridged thick ice. An intersection between the windy open water and the ridged thick ice in the far range probably

causes this. In the near range, ice and open water are clearly separable.

The impression for segment-then-label going from 50 to 10 samples per class for Figure 10.8 (first and second row) is that the change is smaller than for the ML. We notice that the thin ice class (pink) disappears using the fewer samples, for both methods.

Looking at Figure 10.10, it is hard to conclude what amount of samples is actually best (first and second row). With segment-then-label (second coloumn) for 50 samples per image (first row), the ice is classified as flat thick ice and leads, and water is impeccably resolved. But for the 10 samples per class, the open water is partly determined to leads, whereas the ice area is classified to thick ice. This image has a border between ice and water which to more or less extent is classified to leads/newly formed ice in all four classified images.

Using 70% contamination (the examples' page 2, third row):

The resulting images after contamination are plotted for different percentages of contamination. For the examples in Figures 10.6 and 10.10, there was not too much difference in the resulting images, using 20% and 70% contamination. Therefore, only the 70% contamination is shown here. Results for contamination are shown in the third row at the second page of each example.

In Figure 10.6, the contamination does not seem to have any impact on the segment-than label image (compared to the "best" image, Figure 10.5). The only clear difference is the calm open water area that is classified to leads/newly-formed ice. Comparing the "best" ML-classified image with the one using contamination, there is only a small worsening, mostly in the left part of the image where some of the ice is determined as brash ice, and part of the open water as leads. The contamination for the example in Figure 10.10 does not show large changes either. For the ML method, a larger part of the area is classified as brash ice, and areas of open water as leads. The last misclassification also applies to the segment-then-label result.

The contamination effect has a generally larger impact on the example in Figure 10.8. The image with 20% contamination tended to classify more areas to brash ice. In the 70% contamination results, the areas are generally determined to be brash ice, and flat thick ice. This instead of ridged thick ice, as suggested from the "perfect" images.

Using a 70% contamination, we would expect larger distortions in the classified images as a result of the contaminated training data. This is a moment for the discussion.

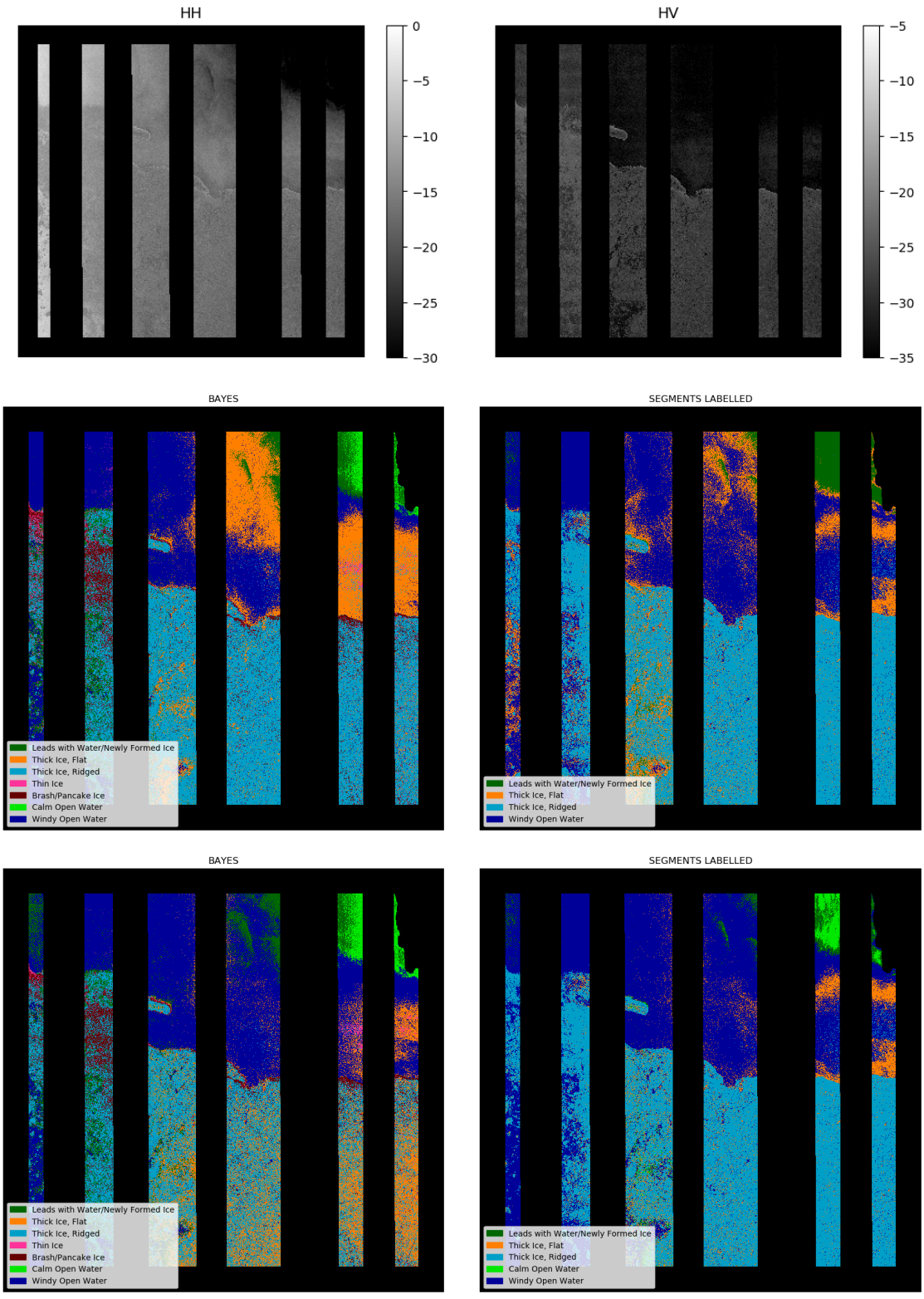


Figure 10.5: Example 1, part 1/2, image no. 15. Row 1: HH and HV log-intensities. Row 2: The classified results based on the full original training data, and row 3: the classified result based on the downsampled training data, for the fully supervised method (coloumn 1) and the segment-then-label method (coloumn 2).

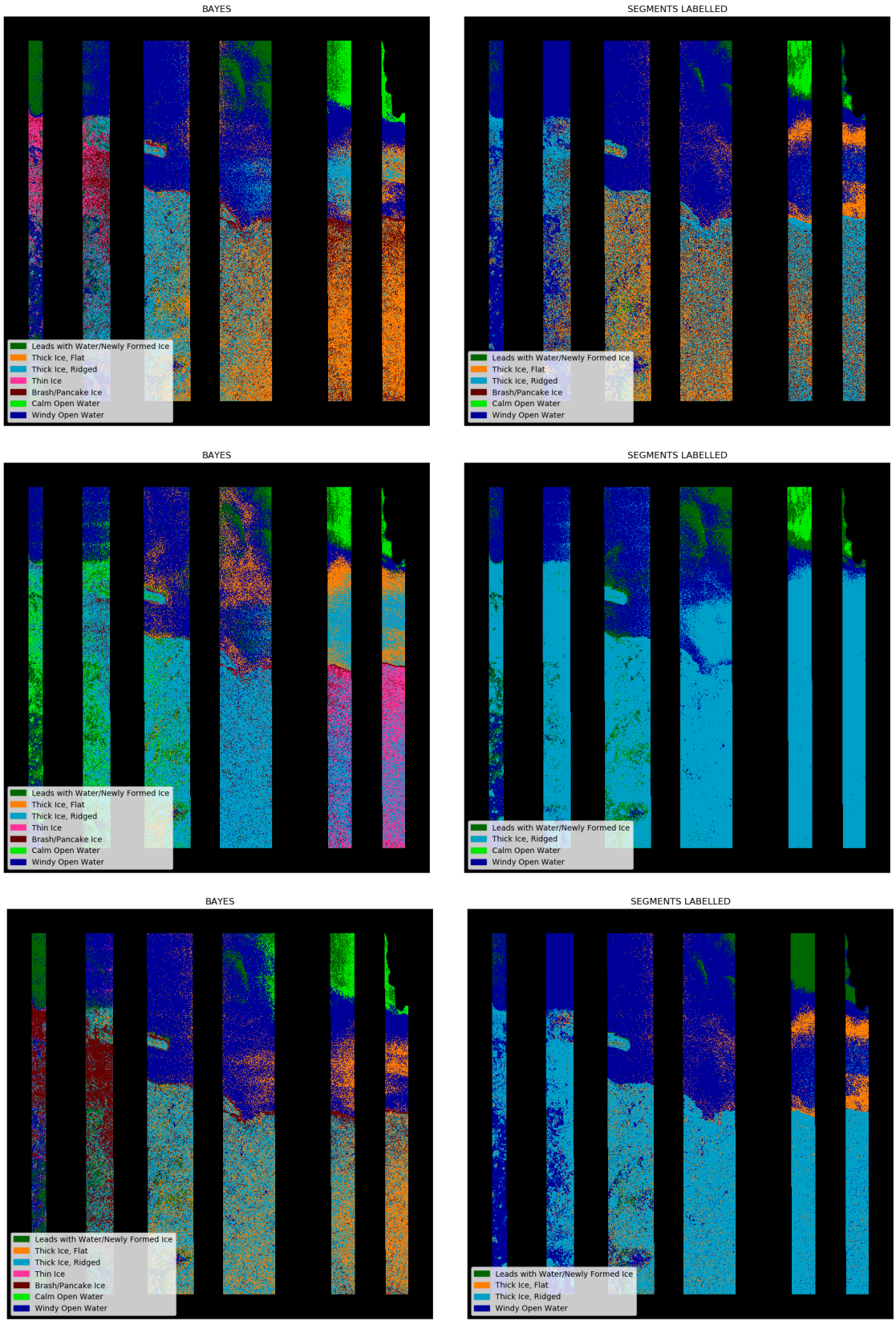


Figure 10.6: Example 1, part 2/2, image no. 15. Coloumn 1: Fully supervised method. Coloumn 2: Segment-then label method. Row 1: 50 samples per class. Row 2: 10 samples per class. Row 3: 70% mislabelling per class.

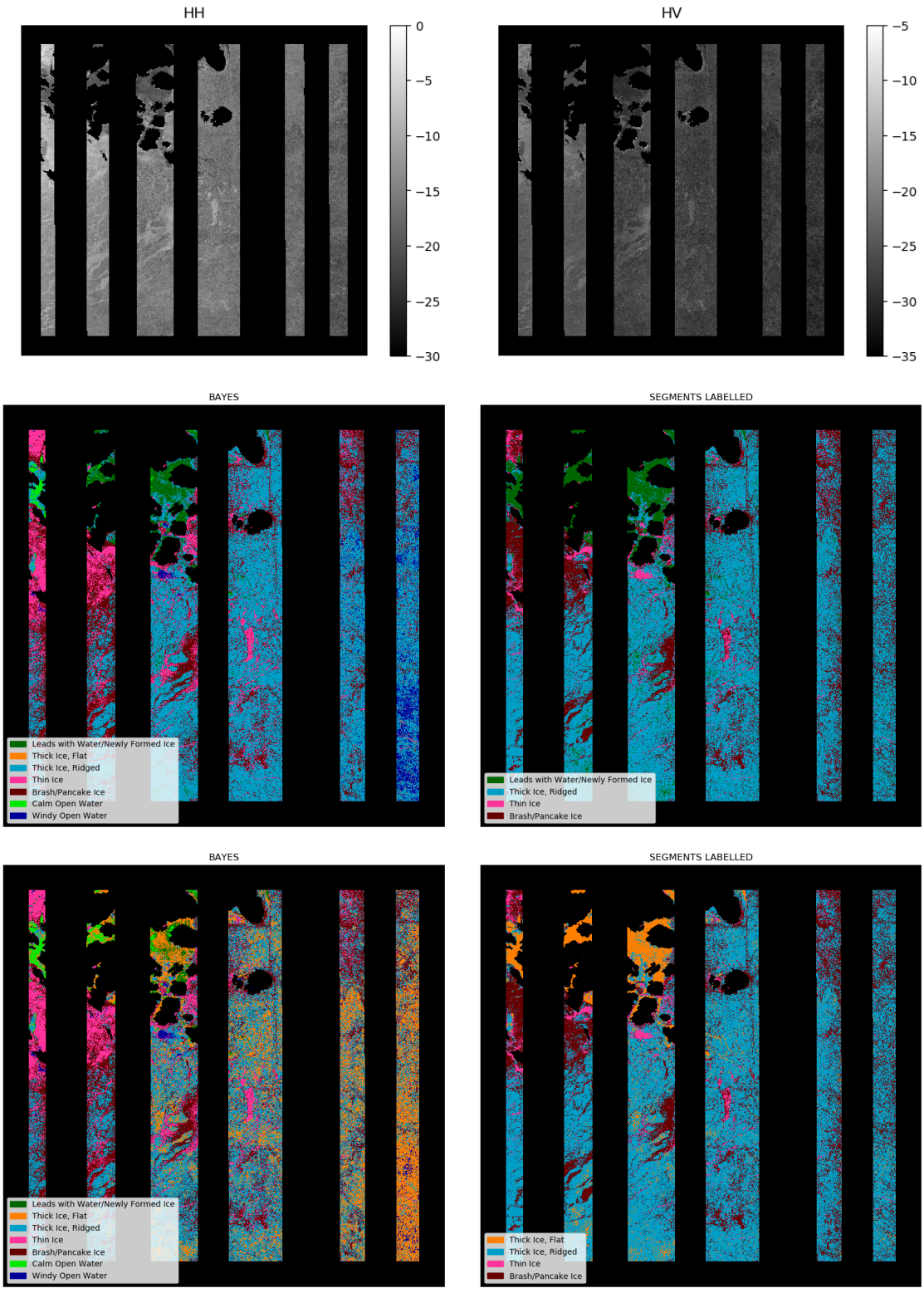


Figure 10.7: Example 2, part 1/2, image no. 2. Row 1: HH and HV log-intensities. Row 2: The classified results based on the full original training data, and row 3: the classified result based on the downsampled training data, for the fully supervised method (coloumn 1) and the segment-then-label method (coloumn 2).

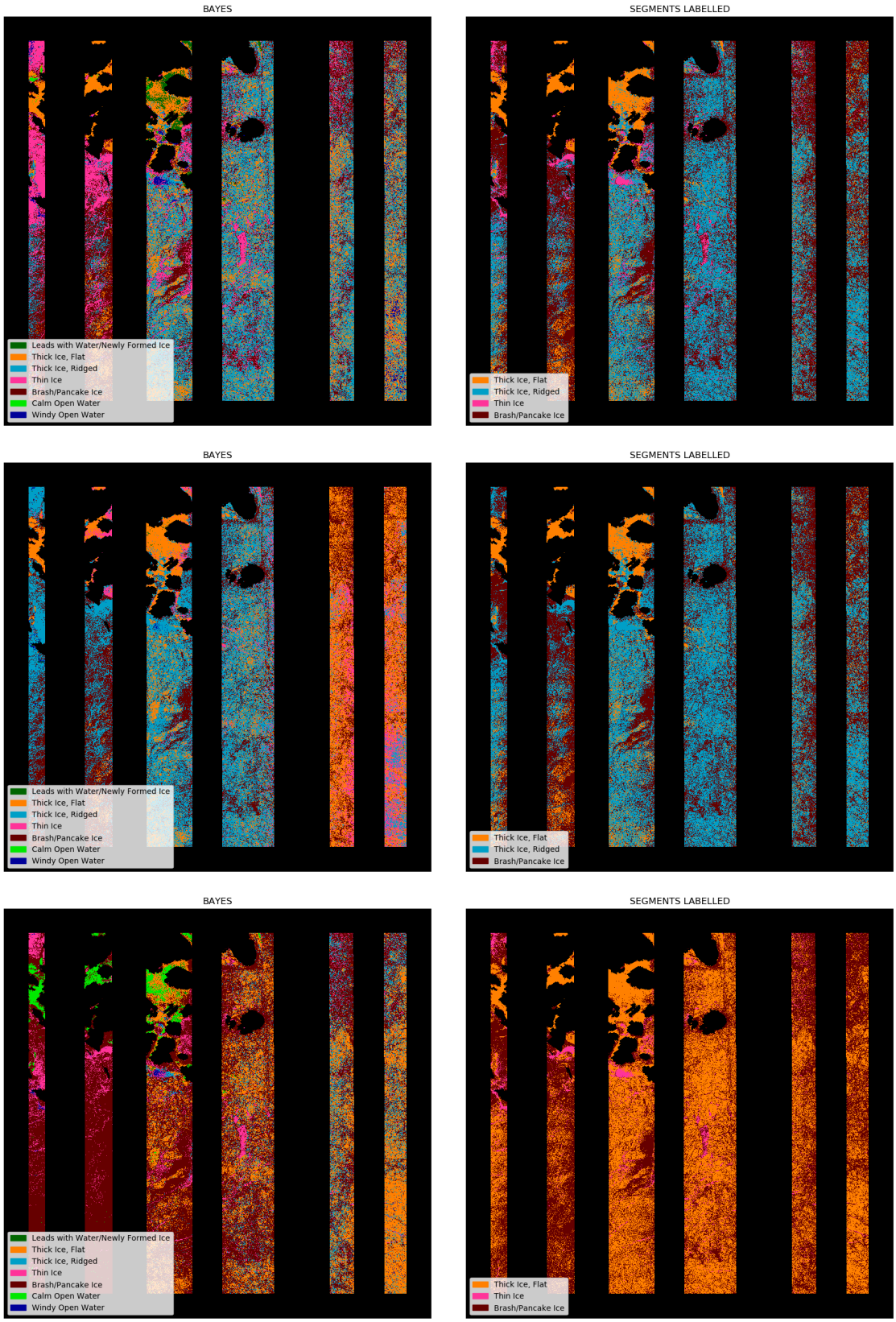


Figure 10.8: Example 2, part 2/2, image no. 2. Coloumn 1: Fully supervised method. Coloumn: Segment-then label method. Row 1: 50 samples per class. Row 2: 10 samples per class. Row 3: 70% mislabelling per class.

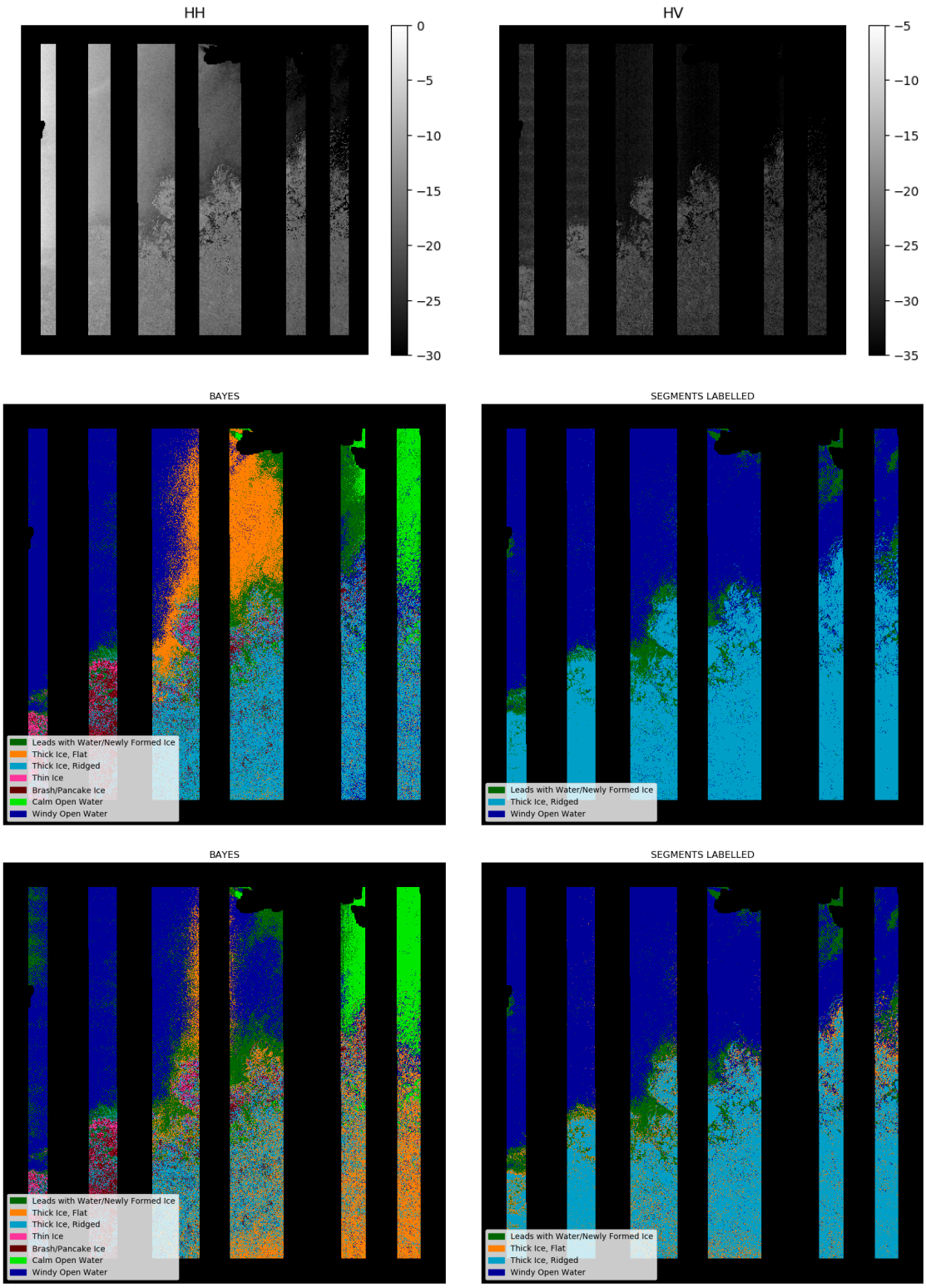


Figure 10.9: Example 3, part 1/2, image no. 20. Row 1: HH and HV log-intensities. Row 2: The classified results based on the full original training data, and row 3: the classified result based on the downsampled training data, for the fully supervised method (coloumn 1) and the segment-then-label method (coloumn 2).

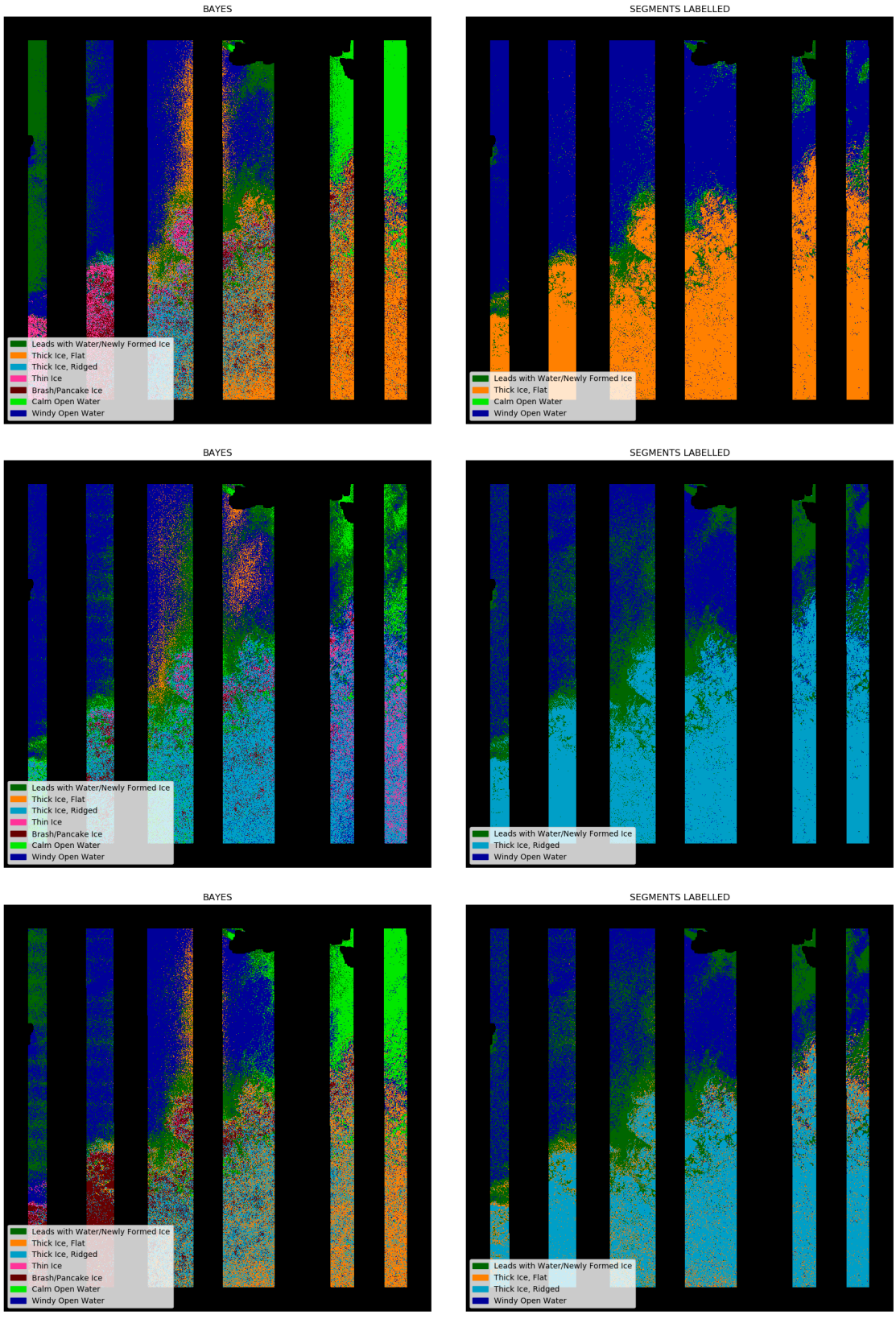


Figure 10.10: Example 3, part 2/2, image no. 20. Coloumn 1: Fully supervised method. Coloumn 2: Segment-then label method. Row 1: 50 samples per class. Row 2: 10 samples per class. Row 3: 70% mislabelling per class.

10.3 Discussion

The results in Figures 10.3 and Figure 10.2 show decaying accuracies. The comparison is on relative accuracy values, being relative to the best result (no contamination and full data set). Ideally, the reference should be the best ever-classified image, a 100%-accuracy classification image. Unfortunately, the best we get is not a perfect classification. The source of the limited accuracy could be limitations in the training data, the particular classes used, or in the methods.

Continuing with the accuracies we have got, we get a accuracy graph for contamination that is too stable and decreases less than expected. The classified images also changes too little with more mislabelling, but there are differences for each image how much mislabelling is necessary for a totally misclassified image. This depends on the image's class composition. After all, 80% contamination in the data means that 80% of the training data has got other labels. The classification can not be unaffected with such large change. A similar accuracy for the 80% contamination case and the no label noise case could not be correct.

The graph for the restriction of training data size seems more reliable (Figure 10.3), even though it also is not increasing much. We see that there is no big difference in going from all our available training data to a subset of 1000 samples per class. The decay starts for less than 1000 samples per class, and is steeper for the fully supervised than for the segment-then-label method. In particular, the decay for less than 100 samples is interesting, as the decay difference becomes more prominent for smaller size. The fully supervised decays more than the segment-than label method in the interval between 100 and 50, but decays even more in the interval between 50 and 10. This is an indication for the segment-then-label method being more robust to small sample sizes, with less than 100 samples for each class. The classified images using 10 and 50 samples per class, substantiate the change rates, and show poorer results for less training data, even though the effect of randomness also is visible by inspecting the scene examples.

In general, one gets the impression that the segment-then-label results are often better looking. They appear smoother, and matches better the distinctions seen in the brightness images. This method better captures the greater image appearance, unlike the ML method, which for the very small number of training data is not able to pick up on it. After all, it does not split up ice classes swath-wise, as is the tendency with less training data for the fully supervised method.

The absent decrease in accuracy for contamination could be due to the training

data and the validation being too similar, even if they are independent. The validation data is taken from the same small polygons as the training data. I.e. inter-polygon pixels may be too similar. It could also be that the training and validation points just happen to be in a small area that looks better in the ML-classified results, even if more real pixels look better elsewhere. If the validation regime is not robust and the validation data are poor, the comparison of the methods could not be done with reliable results.

The training data shown in Figure 6.2, from the polygons within the 5x5 multilooked image, show a quite large variance in brightness values on more of the classes. The large within-class spreads could help explain some of the difficulties. As more contamination is introduced to one class, a pixel with a new label is likely to be within the new label's class' original spread. Thus, this class gets a new point within its old spread, instead of an outlier pixel. No actual contamination is added to this class. Multilooking with a larger filter size (more than 5x5) could be beneficial for reduced variance in each class, such that the contamination would have a clear impact.

The mechanisms for the reduction in size of the data are a bit different. The class variance would also in this case have an impact, in the way that a large variance in a class will make it more likely to achieve a training class slope which is drastically changed. A class with originally little variance would perform better with fewer samples than a class with high variance.

The random sampling per class may have influenced the validation regime, and possibly culminated in poorly represented classes. Experimenting with other sampling methods could be an extension to our work. Sampling in a more structural random fashion could improve both the mean classifier performance and reduce the classification variance, as discussed by Gabrys and Petrakieva (2004) (see Section 1.2.2). E.g. thin ice (class no. 4) and brash ice (class no. 5) have similar incidence angle dependencies (Figure 6.2), but most of the thin ice samples tend to be located in a small incidence angle range. By reducing sample size in a random sampling fashion, we saw that thin ice tended to be misclassified as brash ice, probably as this slope where more likely to remain stable.

In the end, it seems as the training data we are using is not good enough. Using poor training data leads to a method scoring towards a poor solution rather than the actual data distinction. One should consider using a labelled data set which both reflects the actual data distinctions and covers the class variability ranges. Labelling of images requires good knowledge on the surface types, and could be a comprehensive process. This leads us back to the question about the amount of training data available. One of the main reasons for learning unsupervised is reducing the dependency on good training data. Stating the

importance of better training data begs the question.

10.3.1 The expected result

The expected outcome of the classification is first that the segment-then-label method will give higher accuracies than the fully supervised, as the underlying structure of the data is included in the decisions. As the clusters are made before the labelling, the pixels in the same cluster will stay in the same cluster, no matter what training data is used for the labelling. If mislabelled, the whole cluster with all pixels is mislabelled.

The segment-then-label method is expected to be more robust to contamination than the fully supervised. The contamination interchanges labels of the training data, thus the classes will get a certain amount of outlier values, being the values actually belonging to other classes. The contamination changes the slope of the training data classes, such that the class slopes are becoming more and more equal. Still, the ML classifier will compare one pixel against the training data slope for each class. The segment-then-label will compare all pixels within a cluster to the same training data slopes. If the training data slopes change too much, as a consequence of few training data points, or incorrect training data points, it is safer to have the whole segment compared with training data, than only a single point at a time (see point against point, and point against set distances in Section 2.1.1). Though, there is a risk of misclassification of large segments as another consequence.

Regarding the training sample size, the challenge is that the slope will be too biased when the training data points are no longer representative. The training class slopes could change much when going from more to less training data, and the randomness in the data sampling will have an impact on this. As discussed in Section 4.1, using the underlying structure of the data could make up for the lowered representativeness. The segment-then-label method is thus assumed to have a less decreasing accuracy result, which we have also experienced.

10.3.2 Reliability of the results

It is hard to rely on the validation results as long as more contamination does not make the accuracy very much poorer. What happens with more contamination is, that the classes, and their training slopes, are becoming more and more equal. It is likely that we are having a training data set that does not allow for a proper validation regime. This is tested by running the same tests with the opposite training (20%) - validation (80%) split for generalized testing. The

accuracies get lower, but the absent decrease by contamination is not changed significantly.

The amount-restriction graph looks more trustworthy, and it decays with a more reasonable steepness. But, as the contamination graph has clear weaknesses, we also suspect this graph to be negatively impacted, but to a lesser extent.

To confirming our classification and validation scheme, a simulated synthetic dataset could be made and tested in the same comparisons with contamination and data reduction. As one then knows the true data, one could confirm the scoring mechanism to be correct.

11

Conclusion and future work

11.1 Conclusion

This thesis has tested a segment-then-label classification scheme against the supervised Maximum Likelihood classification scheme. The main aims in this thesis were to test and compare the methods for two cases:

- Varying training data sample size
- Varying amount of mislabelling

Both the restrictions of size and mislabelling are done class-wise. The two methods' performances are measured using the mean class accuracy, and have been compared against each other.

The classifications are done to a dataset of Sentinel-1 images on sea ice, and the resulting classifications are presented and compared.

For reduced training data sample size case, the segment-then-label method tends to have a less decaying accuracy than the fully supervised method, especially for small sample sizes. This indicates a larger robustness to smaller sample sizes for the segment-then-label method.

For more contamination the decay is not particularly distinct, probably due to the large within-class variations in the training set. In general, the fully

supervised method gives a higher accuracy, but the segment-then-label method changes less with size and contamination.

We conclude that the segment-then-label method shows tendencies of being more robust to training data size and contamination, but the results are weak. We recommend a further investigation using more reliable training data.

This work is an investigation of the importance of *when* the training data comes into the classification process, either from start in a direct classification, or in a labelling step after involving the underlying data structure.

Limitations

Several limitations to this study need to be acknowledged. Some possible sources of error are:

- The ground truth polygons are made by one (two) persons, and are therefore subject to a subjective opinion.
- Too large within-class variations in the training hide any effects of mislabelling.
- The number of classes in the training data set is limited, just as the number of training samples. Some classes could possibly be split up into subclasses.
- The sampling is done randomly per class, but not per angle.
- The images, and the ground truth drawn from them, are from a large seasonal range.

Even though these sources to error may have degraded the overall results, we assume the same degradation is made to both methods, yielding a fair comparison of the decays.

11.2 Future work

Future research projects could extend this study. First of all, we will mention doing exactly the same as in this work, but using a labelled data set which covers the class variability ranges, and at the same time reflects the actual data distinctions. Possibly, this would require possession of good knowledge on the

surface matter and a thorough labelling effort.

Further topics for extension of the mentioned is development of new labelling strategies, which are not based on distance measures. A possibility is to investigate the confusion matrix for use in the labelling procedure, and eventually the robustness of such a method.

The robustness after involving additional textural features could be explored, as these could help the distinction of classes according to the surface' small geometrical differences.

Part III

Appendix

A

Bayes theory

Let $A_i, i = 1, 2, ...M$ be M events such that $\sum_{i=1}^{M} P(A_i) = 1$. The probability of an event B is then given by the total probability theorem:

$$P(B) = \sum_{i=1}^{M} P(B|A_i)P(A_i)$$

The conditional probability of B given A, P(B|A), is defined as

$$P(B|A) = \frac{P(B, A)}{P(A)}$$

where $P(B, A)$ is the joint probability of event A and B happening.

Bayes rule is the brought up from this to be

$$P(B|A)P(A) = P(A|B)P(B)$$

for which probability components can be replaced by probability density functions when random variables are used:

$$P(\mathrm{x}|A)P(A) = P(A|\mathrm{x})P(\mathrm{x})$$

and

$$P(\mathrm{x}|\mathrm{y})P(\mathrm{y}) = P(\mathrm{y}|\mathrm{x})P(\mathrm{x})$$

where the total probability theorem gives

$$P(\mathrm{x}) = \sum_{i=1}^{M} P(\mathrm{x}|A_i)P(A_i)$$

B

Derivation of the update parameters for the EM-algorithm

The parameter update expressions used in the EM-algorithm's maximisation step is derived here. The concepts needed to get there are explained first.

The joint probability of the observed data x and the unknown labels z, given parameters Θ, can for each sample i be written as

$$p(\mathrm{x_i}, \mathrm{z_i}; \Theta_\mathrm{i})$$

By using the conditional probability sentence, the joint probability can be expanded by conditioning on the unknown labels $\mathrm{z_i}$. The entries in $\mathrm{z_i}$ are assumed to be independent, and the expansion can be written as the product of conditional probabilities for all m.

$$p(\mathrm{x}_i, \mathrm{z}_i; \Theta_i) = \prod_{m=1}^{M} [p(\mathrm{x}_i | z_{im} = 1; \Theta) p(z_{im} = 1)]^{z_{im}}$$

$\mathrm{z_{im}}$ is thought of being 1 for the element corresponding to the selected mixture component m for each sample i, and 0 else; using the superscript z_{im} therefore includes only the for each sample decided components.

The complete log-likelihood is given by

$$L(\theta|x) = \prod_{i=1}^{N} f_i(x|\theta)$$

for N independent distributions f_i. The joint probability from above is inserted for f_i in the likelihood:

$$L(\Theta) = \prod_{i=1}^{N} \prod_{m=1}^{M} [p(\mathrm{x}_i|z_{im} = 1; \Theta)p(z_{im} = 1)]^{z_{im}}$$

The logarithm of the likelihood, or the log-likelihood, is then

$$LL(\Theta) = log\left(\prod_{i=1}^{N} \prod_{m=1}^{M} [p(\mathrm{x}_i|z_{im} = 1; \Theta)p(z_{im} = 1)]^{z_{im}}\right)$$

the logarithm of a product is the sum of the logarithm

$$\begin{aligned} &= \sum_{i=1}^{N} \sum_{m=1}^{M} log\left[p\left(\mathrm{x}_i|z_{im} = 1; \Theta\right) p\left(z_{im} = 1\right)\right]^{z_{im}} \\ &= \sum_{i=1}^{N} \sum_{m=1}^{M} z_{im}\left[log\left(p\left(\mathrm{x}_i|z_{im} = 1; \Theta\right)\right) + log(p(z_{im} = 1))\right] \end{aligned}$$

where $p(z_{im} = 1))$ is the prior for a class and for classification purposes of equiprobable classes may be discarded. $p\left(\mathrm{x}_i|z_{im} = 1; \Theta\right)$ is the Gaussian distribution for each component m for each data point x_i.

The expectation of the log-likelihood above, with respect to Q(**Z**) is given by

$$E\{LL(\Theta)\}_{Q(\mathrm{Z})} = E\left\{\sum_{i=1}^{N} \sum_{m=1}^{M} z_{im}\Big[log\left(p\left(\mathrm{x}_i|z_{im} = 1; \Theta\right)\right)\Big]\right\}$$

the expectation of a sum is the sum of the expectation

$$= \sum_{i=1}^{N} \sum_{m=1}^{M} E\Big\{z_{im}\Big[log\left(p\left(\mathrm{x}_i|z_{im} = 1; \Theta\right)\right)\Big]\Big\}$$

only random variables are subject to the statistical expectation

$$= \sum_{i=1}^{N} \sum_{m=1}^{M} \left(E\left\{z_{im}\right\} log\left(p\left(\mathrm{x}_i | z_{im} = 1; \Theta\right)\right) \right)$$

Inserting the Gaussian distribution gives:

$$\begin{aligned}
&E\{LL(\Theta)\}_{Q(\mathrm{Z})} \\
&= \sum_{i=1}^{N} \sum_{m=1}^{M} \left(E\left\{z_{im}\right\} log\left(\frac{1}{(2\pi)^{d/2} |\Sigma_m|^{1/2}} exp\left(-\frac{1}{2} (x_i - (a_m - b_m \theta_i))^T \Sigma_m^{-1} (x_i - (a_m - b_m \theta_i)) \right) \right) \right) \\
&= \sum_{i=1}^{N} \sum_{m=1}^{M} \left(E\left\{z_{im}\right\} \left(-\frac{d}{2} log(2\pi) - \frac{1}{2} log(|\Sigma_m|) - \frac{1}{2} (x_i - (a_m - b_m \theta_i))^T \Sigma_m^{-1} (x_i - (a_m - b_m \theta_i)) \right) \right)
\end{aligned}$$

where d is the number of dimensions given in the multivariate Gaussian formula.

The maximizing expression for a_m is found by optimizing the expected log-likelihood, inserted the Gaussian density, with respect to a_m. Constants which disappear after derivative are excluded.

$$\frac{\partial E\{LL(\Theta)\}_{Q(Z)}}{\partial \mathrm{a}_m} = \frac{\partial}{\partial \mathrm{a}_m} \sum_{i=1}^{N} E\left\{z_{im}\right\} \left(-\frac{1}{2} (x_i - (a_m - b_m \theta_i))^T \Sigma_m^{-1} (x_i - (a_m - b_m \theta_i)) \right)$$

Using matrix derivative, rule (86) in Petersen and Pedersen (2012). Optimizing

$$= \sum_{i=1}^{N} E\left\{z_{im}\right\} \frac{1}{2} (2\Sigma_m^{-1} (x_i - (a_m - b_m \theta_i))) = 0$$

which gives

$$\Rightarrow \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1} (a_k - b_k \theta_i)) = \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1} x_i$$

such that the maximizing a_m is

$$\hat{a}_m = \frac{\sum_{i=1}^{N} E\left\{z_{im}\right\} x_i + b_k \sum_{i=1}^{N} E\left\{z_{im}\right\} \theta_i}{\sum_{i=1}^{N} E\left\{z_{im}\right\}}$$

The maximizing expression for b_m is found by optimizing the expected log-likelihood, inserted the Gaussian density, with respect to b_m. Constants which disappear after derivative are excluded.

$$\frac{\partial E\{LL(\Theta)\}_{Q(Z)}}{\partial \mathrm{b}_m} = \frac{\partial}{\partial \mathrm{b}_m} \sum_{i=1}^{N} E\left\{z_{im}\right\} \left(-\frac{1}{2}(x_i - (a_m - b_m\theta_i))^T \Sigma_m^{-1}(x_i - (a_m - b_m\theta_i))\right)$$

Using the matrix derivative, rule (86) in Petersen and Pedersen (2012). Optimizing.

$$= \sum_{i=1}^{N} E\left\{z_{im}\right\} \frac{1}{2}(2\Sigma_m^{-1}(x_i - (a_m - b_m\theta_i))\theta_i) = 0$$

which gives

$$\Rightarrow \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1}(a_k - b_k\theta_i)\theta_i) = \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1} x_i$$

such that the maximizing b_m is

$$\hat{b}_m = \frac{a_k \sum_{i=1}^{N} E\left\{z_{im}\right\} \theta_i - \sum_{i=1}^{N} E\left\{z_{im}\right\} x_i}{\sum_{i=1}^{N} E\left\{z_{im}\right\} \theta_i^2}$$

The maximizing expression for Σ_m is also found by optimizing the expected log-likelihood with respect to the parameter itself. The derivative, inserted with Gaussian probability density without prior, and without terms that vanish after derivative is taken.

$$\frac{\partial E\{LL(\Theta)\}_{Q(Z)}}{\partial \Sigma_m}$$
$$= \frac{\partial}{\partial \Sigma_m} \sum_{i=1}^{N} \left(E\left\{z_{im}\right\} \left(-\frac{1}{2} log(|\Sigma_m|) - \frac{1}{2}(x_i - (a_m - b_m\theta_i))^T \Sigma_m^{-1}(x_i - (a_m - b_m\theta_i))\right)\right)$$

Using matrix derivative rules (72) and (141) and (396) in Petersen and Pedersen (2012) , and then optimizing

$$= \sum_{i=1}^{N} E\left\{z_{im}\right\} \frac{1}{2}\left(-\Sigma_m^{-1} + \Sigma_m^{-1}(x_i - (a_m - b_m\theta_i))(x_i - (a_m - b_m\theta_i))^T (\Sigma_m^{-1})^T\right) = 0$$

gives

$$\Rightarrow \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1} = \sum_{i=1}^{N} E\left\{z_{im}\right\} \Sigma_m^{-1}(x_i - (a_m - b_m\theta_i))(x_i - (a_m - b_m\theta_i))^T (\Sigma_m^{-1})^T$$

All covariance matrices are symmetric and positive semi-definite. A symmetric matrix A equals its transpose: $A = A^T$. This gives the final maximizing expression

$$\hat{\Sigma}_m = \frac{\sum_{i=1}^{N} E\{z_{im}\}(x_i - (a_m - b_m\theta_i))(x_i - (a_m - b_m\theta_i))^T}{\sum_{i=1}^{N} E\{z_{im}\}}$$

E-step: The update expression for z_{im} is derived in the following way:

$$E\{z_{im}\}_{p(Z|X;\Theta)} = p(z_{im} = 1|x_i; \Theta)$$

Using the sentence about conditional probability

$$= \frac{p(z_{im} = 1, x_i|\Theta)}{p(x_i|\Theta)}$$

Using conditional probability again, conditioning on z_{im}

$$= \frac{p(x_i|z_{im} = 1, \Theta)p(z_{im} = 1)}{p(x_i|\Theta)}$$

Using the rule of total probability in the denominator

$$= \frac{p(x_i|z_{im} = 1, \Theta)p(z_{im} = 1)}{\sum_{k=1}^{M} p(x_i|z_{ik} = 1, \Theta)P(z_{ik} = 1)}$$

Recognizing the probability $p(z_{im} = 1)$ to be the prior, which we discard due to equiprobability. The final expression

$$E\{z_{im}\}_{p(Z|X;\Theta)} = \frac{p(x_i|z_{im} = 1, \Theta)}{\sum_{k=1}^{M} p(x_i|z_{ik} = 1, \Theta)}$$

C

Tables

Table C.1: Enumerated image scenes used in the thesis

Nr	Image
1	S1A_EW_GRDM_1SDH_20150328T105745_20150328T105846_005231_0069B3_D34B
2	S1A_EW_GRDM_1SDH_20150327T115532_20150327T115632_005217_00696B_D22F
3	S1A_EW_GRDM_1SDH_20150504T163454_20150504T163554_005774_0076AC_3188
4	S1B_EW_GRDM_1SDH_20170716T122837_20170716T122941_006513_00B73D_BB34
5	S1A_EW_GRDM_1SDH_20150331T112258_20150331T112358_005275_006ABB_CD7F
6	S1A_EW_GRDM_1SDH_20150412T112159_20150412T112259_005450_006F2D_D076
7	S1A_EW_GRDM_1SDH_20160427T141524_20160427T141624_011008_0108BB_763B
8	S1A_EW_GRDM_1SDH_20160604T171627_20160604T171727_011564_011A8C_4390
9	S1A_EW_GRDM_1SDH_20160709T131853_20160709T131953_012072_012AC1_D788
10	S1A_EW_GRDM_1SDH_20160711T161922_20160711T162022_012103_012BC2_0B78
11	S1A_EW_GRDM_1SDH_20150406T152850_20150406T152950_005365_006CEC_6952
12	S1A_EW_GRDM_1SDH_20160708T141528_20160708T141628_012058_012A49_CA0E
13	S1A_EW_GRDM_1SDH_20160709T163445_20160709T163542_012074_012ACF_52EF
14	S1A_EW_GRDM_1SDH_20160722T171529_20160722T171629_012264_0130ED_CD79
15	S1A_EW_GRDM_1SDH_20160711T161822_20160711T161922_012103_012BC2_A3B8
16	S1A_EW_GRDM_1SDH_20150412T112259_20150412T112359_005450_006F2D_AC40
17	S1A_EW_GRDM_1SDH_20150508T160153_20150508T160253_005832_007813_7CF7
18	S1A_EW_GRDM_1SDH_20160614T141527_20160614T141627_011708_011F14_5612
19	S1A_EW_GRDM_1SDH_20160627T163444_20160627T163544_011899_012521_5B2A
20	S1A_EW_GRDM_1SDH_20160709T131753_20160709T131853_012072_012AC1_E527
21	S1A_EW_GRDM_1SDH_20150620T141524_20150620T141624_006458_0088D5_773C
22	S1A_EW_GRDM_1SDH_20160627T163544_20160627T163644_011899_012521_8539
23	S1A_EW_GRDM_1SDH_20160608T164324_20160608T164424_011622_011C50_DA37
24	S1A_EW_GRDM_1SDH_20160708T141628_20160708T141728_012058_012A49_AE8A
25	S1A_EW_GRDM_1SDH_20160708T173158_20160708T173258_012060_012A5D_565D
26	S1A_EW_GRDM_1SDH_20160513T165924_20160513T170024_011243_011024_564C
27	S1A_EW_GRDM_1SDH_20150716T153742_20150716T153842_006838_009369_9351

Table C.2: Training data samples per class per image for the downsampled ground truth. Image numbers according to Table C.1. Down. denotes the whole downsampled image. Masked denotes the samples still present after borders and between-swath areas are masked away. The masked samples are used for training.

	cl1		cl2		cl3		cl4		cl5		cl6		cl7	
Im.	Down.	Masked	Down.	Masked	Down.	Masked	Down.	Masked	Down.	Masked	Down.	Masked	Down.	Masked
1	153	29	837	430	542	0	1.058	831	0		0		0	
2	5.232	2578	0		1.901	0	291	269	0		0		0	
3	269	117	0		0		0		0		0		0	
4	1.014	74	0		3.220	380	0		0		0		0	
5	41	41	372	372	240	240	443	443	0		0		0	
6	829	429	711	59	237	0	177	0	0		0		0	
7	8.011	1.556	235	235	12.709	9.412	4.570	3.696	0		0		0	
8	9.950	1.456	0		5.233	2.997	0		0		0		0	
9	52	21	70	26	150	0	0		0		0		0	
10	15	0	142	142	59	59	0		0		0		0	
11	311	181	0		4.137	4.137	1.522	479	16.620	7.763	50.750	32.625	744.363	255.168
12	0		0		24.979	7.939	0		4.914	3.101	187.138	48.940	645.198	288.022
13	10.844	1.349	0		27.763	17.520	0		10.928	3.075	141.441	25.858	519.261	206.956
14	9.482	1.952	0		18.198	9.714	0		4.295	4.047	538.639	168.585	0	
15	0		0		10.316	6.106	0		0		245.731	47.831	943.716	394.917
16	40	0	0		0		0		0		8.461	1.080	93.787	31.285
17	109	5	0		0		132	74	0		5.018	0	692.751	282.826
18	0		0		5.475	4.690	0		0		39729	13.730	148.393	82.274
19	332	288	281	281	0		0		0		100.227	19.519	0	
20	0		0		4.795	3.955	0		0		99.536	28.847	1.192.838	519.486
21	88	16	0		0		0		0		132.430	32.413	0	
22	12.295	6.561	336	0	38.081	20.508	0		0		0		0	
23	36.056	24.828	0		2.358	1.847	0		0		0		0	
24	31	10	100	53	1.037	0	0		0		0		0	
25	14.424	0	0		0		0		0		0		0	
26	0		0		0		0		0		0		52.374	798
27	0		0		0		0		0		0		62.020	0
Sum	109.578	45.491	3.084	1.598	161.431	89.504	8.193	5.792	36.757	17.986	1.549.100	419.428	5.094.701	2.061.732

Bibliography

Barber, D. G., Yackel, J. and Hanesiak, J. (2001), 'Sea ice, RADARSAT-1 and arctic climate processes: A review and update', *Canadian journal of remote sensing* **27**(1), 51–61.

Bazi, Y. and Melgani, F. (2010), 'Gaussian process approach to remote sensing image classification', *IEEE transactions on geoscience and remote sensing* **48**(1), 186–197.

Bliss, A. C., Steele, M., Peng, G., Meier, W. N. and Dickinson, S. (2019), 'Regional variability of arctic sea ice seasonal change climate indicators from a passive microwave climate data record', *Environmental Research Letters* .

Campbell, J. B. and Wynne, R. H. (2011), *Introduction to remote sensing*, Guilford Press.

Canada Centre for Mapping and Earth Observation, Natural Resources Canada (2015), 'Radar polarimetry'. Modified: 2015-11-20, Accessed: 2019-04-17.
URL: *https://www.nrcan.gc.ca/earth-sciences/geomatics/satellite-imagery-air-photos/satellite-imagery-products/educational-resources/9275*

Casey, J. A., Howell, S. E., Tivy, A. and Haas, C. (2016), 'Separability of sea ice types from wide swath C-and L-band synthetic aperture radar imagery acquired during the melt season', *Remote sensing of environment* **174**, 314–328.

Chawla, N. V. and Karakoulas, G. (2005), 'Learning from labeled and unlabeled data: An empirical study across techniques and domains', *Journal of Artificial Intelligence Research* **23**, 331–366.

Collecte Localisation Satellites (CLS), ESA (2016), 'Sentinel-1 Product Definition'.
URL: *https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/document-library/-/asset_publisher/1dO7RF5fJMbd/content/sentinel-1-product-definition*

Copernicus Space Component Mission Management Team (2018), 'Sentinels High Level Operations Plan (HLOP)'. Type: PL, Issue/revision: 2/2.
URL: *https://sentinel.esa.int/documents/247904/685154/Sentinel_High_Level_Operations_Plan*

Cristea, A. and Doulgeris, A. (2018), 'Integrating incidence-angle dependence in the segmentation of UAVSAR images'. UiT. Poster Presentation at SeaSar.

Cristea, A., van Houtte, J. and Doulgeris, A. (2019), 'Automatic segmentation of SAR images considering the incidence angle effect', *IEEE Transactions on Geoscience and Remote Sensing* . Under revision.

Cutler, P. J., Schwartzkopf, W. C. and Koehler, F. W. (2015), Robust automated thresholding of SAR imagery for open-water detection, *in* 'Radar Conference (RadarCon), 2015 IEEE', IEEE, pp. 0310–0315.

Doulgeris, A. P. (2013), 'A simple and extendable segmentation method for multi-polarisation SAR images'. POLinSAR in Frascati.

Doulgeris, A. P. (2015), 'An Automatic U-Distribution and Markov Random Field Segmentation Algorithm for PolSAR Images', *IEEE Transactions on Geoscience and Remote Sensing* **53**(4), 1819–1827.

Doulgeris, A. P. and Cristea, A. (2018), Incorporating Incidence Angle Variation into SAR Image Segmentation, *in* 'IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium', IEEE, pp. 8543–8546.

Doulgeris, A. P. and Eltoft, T. (2010), 'Scale mixture of Gaussian modelling of polarimetric SAR data', *EURASIP Journal on Advances in Signal Processing* **2010**, 2.

Elachi, C. and Van Zyl, J. J. (2006), *Introduction to the physics and techniques of remote sensing*, Vol. 28, John Wiley & Sons.

Fang, Y., Xu, L., Peng, J., Yang, H., Wong, A. and Clausi, D. A. (2018), 'Unsupervised Bayesian Classification of a Hyperspectral Image Based on the Spectral Mixture Model and Markov Random Field', *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (99), 1–13.

Friedman, J., Hastie, T. and Tibshirani, R. (2001), Vol. 1, Springer series in statistics New York.

Gabrys, B. and Petrakieva, L. (2004), 'Combining labelled and unlabelled data in the design of pattern classification systems', *International journal of ap-*

proximate reasoning **35**(3), 251–273.

Ghahramani, Z. (2004), Unsupervised Learning, *in* O. Bousquet, U. von Luxburg and G. Rätsch, eds, 'Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures', Springer, pp. 72–112.

Haykin, S., Lewis, E. O., Raney, R. K. and Rossiter, J. R. (1994), *Remote sensing of sea ice and icebergs*, Vol. 13, John Wiley & Sons.

Koltunov, A. and Ben-Dor, E. (2001), 'A new approach for spectral feature extraction and for unsupervised classification of hyperspectral data based on the Gaussian mixture model', *Remote Sensing Reviews* **20**(2), 123–167.

Lyons, T. and Arribas, I. P. (2018), 'Labelling as an unsupervised learning problem', *arXiv preprint arXiv:1805.03911* .

Maggiori, E., Tarabalka, Y., Charpiat, G. and Alliez, P. (2017), 'Convolutional neural networks for large-scale remote-sensing image classification', *IEEE Transactions on Geoscience and Remote Sensing* **55**(2), 645–657.

Mäkynen, M. and Karvonen, J. (2017), 'Incidence angle dependence of first-year sea ice backscattering coefficient in SENTINEL-1 SAR imagery over the Kara Sea', *IEEE Transactions on Geoscience and Remote Sensing* **55**(11), 6170–6181.

Mäkynen, M., Manninen, A. T., Simila, M., Karvonen, J. A. and Hallikainen, M. T. (2002), 'Incidence angle dependence of the statistical properties of C-band HH-polarization backscattering signatures of the Baltic Sea ice', *IEEE Transactions on Geoscience and Remote Sensing* **40**(12), 2593–2605.

Moen, M.-A., Anfinsen, S., Doulgeris, A., Renner, A. and Gerland, S. (2015), 'Assessing polarimetric sar sea-ice classifications using consecutive day images', *Annals of Glaciology* **56**(69), 285–294.

Ochilov, S. and Clausi, D. A. (2010), Automated classification of operational sar sea ice images, *in* '2010 Canadian Conference on Computer and Robot Vision', IEEE, pp. 40–46.

Ochilov, S. and Clausi, D. A. (2012), 'Operational sar sea-ice image classification', *IEEE Transactions on Geoscience and Remote Sensing* **50**(11), 4397.

Onstott, R. and Shuchman, R. (2004), SAR Measurements of Sea Ice, *in* C. R. Jackson and J. R. Apel, eds, 'Synthetic aperture radar: marine user's manual', chapter 3, pp. 81–115. URL: `http://www.sarusersmanual.com/ManualPDF/`

NOAASARManual_CH03_pg081_116.pdf. Accessed: 2019-05-11.

Park, J.-W., Kim, H.-C., Hong, S.-H., Kang, S.-H., Graber, H. C., Hwang, B. and Lee, C. M. (2016), 'Radar backscattering changes in Arctic sea ice from late summer to early autumn observed by space-borne X-band HH-polarization SAR', *Remote Sensing Letters* **7**(6), 551–560.

Park, J.-W., Korosov, A. A., Babiker, M., Sandven, S. and Won, J.-S. (2018), 'Efficient thermal noise removal for Sentinel-1 TOPSAR cross-polarization channel', *IEEE Transactions on Geoscience and Remote Sensing* **56**(3), 1555–1565.

Parshakov, I., Coburn, C. and Staenz, K. (2014*a*), 'Automated class labeling of classified landsat tm imagery using a hyperion-generated hyperspectral library', *Photogrammetric Engineering & Remote Sensing* **80**(8), 797–805.

Parshakov, I., Coburn, C. and Staenz, K. (2014*b*), Z-score distance: A spectral matching technique for automatic class labelling in unsupervised classification, *in* 'Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International', IEEE, pp. 1793–1796.

Petersen, K. B. and Pedersen, M. S. (2012), 'The Matrix Cookbook'. Version: November 15, 2012. Accessed: 2019-05-20.
URL: *https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf*

Qin, A. and Clausi, D. A. (2010), 'Multivariate image segmentation using semantic region growing with adaptive edge penalty', *IEEE Transactions on Image Processing* **19**(8), 2157–2170.

Ressel, R., Singha, S., Lehner, S., Rösel, A. and Spreen, G. (2016), 'Investigation into different polarimetric features for sea ice classification using X-band synthetic aperture radar', *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens* **9**(7), 3131–3143.

Scheuchl, B., Caves, R., Cumming, I. and Staples, G. (2001), Automated sea ice classification using spaceborne polarimetric sar data, *in* 'IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217)', Vol. 7, IEEE, pp. 3117–3119.

Sentinel-1 Mission Performance Centre (MPC) (2017*a*), 'Sentinel-1 Level 1 Detailed Algorithm Definition'.
URL: *https://sentinel.esa.int/documents/247904/1877131/Sentinel-1-Level-1-Detailed-Algorithm-Definition*

Sentinel-1 Mission Performance Centre (MPC) (2017*b*), 'Thermal denoising of products generated by the Sentinel-1 IPF'. Issue: 1.1. Reference: MPC-0392. Type: Technical Document. Updated: 2017-11-28.
URL: *https://sentinel.esa.int/documents/247904/2142675/Thermal-Denoising-of-Products-Generated-by-Sentinel-1-IPF*

Soh, L.-K., Tsatsoulis, C., Gineris, D. and Bertoia, C. (2004), 'ARKTOS: An intelligent system for SAR sea ice image classification', *IEEE Transactions on geoscience and remote sensing* **42**(1), 229–248.

Theodoridis, S. and Koutroumbas, K. (2009), *Pattern recognition*, Vol. 1, 4 edn, Elsevier Inc.

WMO Sea-ice nomenclature (2017). Edition 1970-2017. Nr 259. Vol. I, II and III.
URL: *http://www.aari.ru/gdsidb/xml/wmo_259.php*

Yu, P., Qin, A. and Clausi, D. A. (2012), 'Feature extraction of dual-pol SAR imagery for sea ice image segmentation', *Canadian Journal of Remote Sensing* **38**(3), 352–366.

Yu, Q. and Clausi, D. A. (2008), 'IRGS: Image segmentation using edge penalties and region growing', *IEEE transactions on pattern analysis and machine intelligence* **30**(12), 2126–2139.

Zakhvatkina, N., Smirnov, V. and Bychkova, I. (2019), 'Satellite SAR Data-based Sea Ice Classification: An Overview', *Geosciences* **9**(4), 152.

Zakhvatkina, N. Y., Alexandrov, V. Y., Johannessen, O. M., Sandven, S. and Frolov, I. Y. (2013), 'Classification of sea ice types in ENVISAT synthetic aperture radar images', *IEEE Transactions on Geoscience and Remote Sensing* **51**(5), 2587–2600.