Visual keras package

Weights biases, training stabilität

Patch Extraction and CNN

Train with smaller resnet backbone

* Try without batchnorm
* Add per class IOU
* 5-crossfold in all tests
* More epochs?
* Augmentation modes change
* Weights and biases
* Fine-tune freeze half
* Add timer
* Track pixel accuracy and F1, too (background class contained?)
* (patch extraction: use sliding window moved by a random stride between 10% and 40% of the patch size K)
* Check if patch extraction before or after split

“The main contributions of this thesis are the following:

…

On top of that, we performed preliminary experiments for fusing VIS and IR. A new dataset of labelled IR images is created.

The reminder of this thesis is organized as follows:

Run again:

* Mode 0 (changed cropping to rotation)
* Mode 2 (change parameter random brightness contrast)
* Mode 4 (added sharpen blurr motion blurr)
* Test run offline augmentation again
* So far, in on the fly augmentation, augmentation is applied to almost each image! (set p=0.5)

CHECK WHAT ON FLY DOES

* Wichtig: Wie wähle ich das ‚beste‘ Modell aus (nach einer bestimmten Anzahl von Epochen = Early stopping, bei konvergenz averagen…? Validierungsdatensatz ist ungesehen, also quantitative auswertung darauf ok)
* Auf mehr epochen trainieren, v.a. bei augmentierung wo noch instabil
* Per class iou exemplarisch ausgeben lassen
* Vergleich der melt pond fraction optisch – IR
* Dropout adden um overfitting zu minimieren
* Change implementation of mode 0 and mode 1
* Offline augmentierung: model konvergiert schneller, da quasi eine epoche zwanzig mal so lang ist
* Wie wird decoder auf backbone gestackt (wo im backbone? Letzte convolutional layer?)
* Wie wird mean iou score berechnet (auf alle drei klassen oder nur melt pond und sea ice
* Instabilität zwischen verschiedenen trainings kommen durch stochastische natur: zufällige initialisierung, daten werden in zufälliger reihenfolge hineingegeben (stochastic gradient descent)
* Wird average crossfold auf letzte epoche gerechnet?
* Wie genau findet das pretraining statt ( layer freezen, decoder neu…?)
* Wird der datensatz bei on fly augmentierung auch vergrößert?
* Patch size: wenn zu klein (32, 64) gibt es randprobleme, wenn zu groß (480) Probleme mit kleineren Eisschollen
* Generell: viel overfitting (Unterschied training – test). Weniger wenn mit augmentierung
* Batch size und learning rate haben miteinander zu tun: Wenn kleinere batch size tendenziell kleinere learning rate. V.a. für training from scratch interessant, bei imagenet eigentlich nur für decoder relevant.
* Einleitung: mehr auf energy budget / klimamodelle fokussieren / hervorheben. Hannah zitieren
* Final: Größerer trainingsdatensatz notwendig (auch concerning schwankungen in trainingsstabilität

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The Arctic region is warming faster than the rest of the world. During the summer months, sea ice begins to melt and pools of water form on its surface. These melt ponds can cover up to about 50-60% of the sea ice surface as the season progresses. Melt ponds are of major importance for the Arctic energy budget because they are darker than snow and ice and absorb significantly more sunlight. This leads to further melting of surrounding areas and rapidly changing surface structures. Melt ponds also provide habitat for various organisms and change light transmission into ocean water, with implications for the underlying marine ecosystem.

Observations of melt ponds are hampered by the remoteness of the Arctic Ocean. Melt ponds have been investigated using ground-based measurements. However, these methods are locally limited to small areas and do not allow a realistic representation of the entire Arctic. Remote sensing techniques collect data at a distance, which makes it possible to retrieve melt ponds on a larger scale. Satellite images can cover major parts of the Arctic on a regular basis, but they often lack resolution. In low-resolution satellite imagery, one pixel can contain many ponds, making accurate pond retrieval a challenge. High-resolution satellite imagery of the Arctic is only available for certain spectral bands (WORK ON THIS).

Remote sensing retrievals are characterized by (1) spatial resolution and (2) spectral coverage.

Quantifying the distribution of melt ponds is crucial for studying the Arctic sea ice dynamics and ecosystem impacts. Accurate estimates of melt pond fraction is required for models that simulate and predict the Arctic climate system, which collectively contribute to our understanding of long-term trends and climate change impacts. Melt pond fraction, the proportion of sea ice surface covered by melt ponds, can be derived from image data. Methods are needed that separate images into melt pond, sea ice and ocean classes.

This thesis presents the development of a Neural Network that automatically segments IR helicopter images into sea ice, ocean and melt pond classes. The following thesis is structured as follows:…

…overall leading to a better understanding of Arctic sea ice dynamics and ecosystem impacts.

* Include: melt pond optical properties and IR difficulties (Remote sensing techniques 🡪 how melt pond images look like, IR and VIS)

Semantic segmentation is the task of assigning a class value to each pixel of an image. Simple automated methods can be divided into pixel-based methods and those that take semantic context into account. Pixel-based methods mainly include thresholding, setting a color threshold. Broader methods include edge-based segmentation that rely of color gradients and object occlusion methodology and texture based methods.

Previous work in melt pond segmentation mainly comprised working with optical images. Thresholding and edge based methods.

Temporally and spatially varying temperatures make methods not applicable. Method must be larger context and shape of surface structures into account. In recent year, CNN….

CNNs are supervised classification methods that are inspired by the brain and learn image patterns by… When extended with a decoder can be used to reproduce location and used for segmentation tasks.

One disadvantages of CNNs is the reuirement of large amounts of training data. CNN comprise millions of parameters that are tuned during learning. When only small amount of data is shown, model is prone to overfitting and lack of generizability (references).

On the other hand, labelling training data for segmentation tasks is very expensive in terms of human effort. Methods that account for small training dataset include (1) synthetic dataset extension and (2) transfer learning.

Synthetic dataset extension

Patch Extraction

Data Augmentation

Small training sets can be synthetically enlargened by modification to the image.

Transfer Learning

Transfer Learning means copying parts of training data from a different task or domain for the task at hand. This can be used then to extend to the new task and has been shown effectively for small datasets.

By encorporating careful strategies, small dataset training is possible and led to successful results in the past

**Methodology**

Unet

* Describe original Unet and what is different in sm implementation

Semantic segmentation in melt pond research, divide in CV: thresholding, edge-based, color-based, supervised (Random forest, combines color and texture, but generalization is often bad)

Why they are not applicable.

CNN and why they are good.

Why CNNs are bad with small training datasets.

How this can be tackled:

(1) Augmentation: What is augmentation. What methods do exist (refer to other studies, eg geometrical transformation…), maybe to remote sensing. Why it is successful. Say which ones I want to use first. Refer to DL and GAN approaches for further research.

(2) Pretraining / Transfer Learning: training a network on a big dataset and then using weights as initial weights (usually just weights of CN, effective since many image datasets share low-level spatial characteristics that are better learned with big data. Pretraining still enables flexibility in network architecture desing

(3) Patch sizes

Which methods this thesis is trying to apply and compare

[1] Dataset Growth in Medical Image Analysis Research (Landau)

[2] Deng J, Dong W: ImageNet: …. 2009

[3] Shirke 2018: Drop:…

[4] Palatucci, Pomerleau: Zero-shot learning … 2009

[5] Xian: Zero-shot learning 2019

**[6] Shorten: A survey on image data augmentation for DL, 2019**

[7] Naveed: Image Mixing and deleting for data augmentation 2021

[8] Khosla Enhancing performance of DL models with different data augmentation techniques… 2020

[19] Shijie Ping Peiyi: Research on data augmentation for image classification based on CNN 2017

[23] Krizhevsky Learning multiple layers of features from tiny images 2009

**Regularization:**

Look references in shorten et al

* General:
* Dropout:
* Transfer Learning:
* Pretraining:
* One-shot zero-shot:

**Augmentation Paper Shorten**

[A survey on Image Data Augmentation for Deep Learning (springeropen.com)](https://journalofbigdata.springeropen.com/counter/pdf/10.1186/s40537-019-0197-0.pdf)

* No big data 🡪 overfitting. Overfitting = network learns a function with very high variance such as to perfectly model the training data.
* Divided into **Data Warping** (transform existing images such that label is preserved) and **Oversampling** (create synthetic instances and add them to the training set: mixing images, feature space augmentations, GANs)

**Augmentation Paper Johanna**

Key takeaways

Data Augmentation is needed because networks contain millions of parameters 🡪 it is needed to show a proportional amount of examples. And the number of parameters you need is proportional to the complexity of the task that the model performs.

Later: Maybe use GAN to translate summer images into winter domain

* Small training dataset 🡪 lack of generizability and tendency to overfit. CNN: “ability to preserve spatial properties of images due to highly parameterized and sparsely connected kernels. Spatial resolution is systematically downsampled, while the depth of the feature maps is simultaneously expanded 🡪 network learns relatively low-dimensional yet powerful representations that […] greatly surpass the effectiveness of handcrafted features.”
* “Nearly every task domain benefitting from computer vision publishes new research reporting results using CNN as a significant component in novel systems”.
* “CNNs are prone to overfit on small datasets because of their massive numbers of parameters. Overfitting occurs when the network perfectly models the training set but cannot generalize its learning to predict the class of unseen data accurately. The overfitting problem has generated a need and an expectation for large data sets and is one of the pressures escalating data size growth. Data size is currently associated with research quality: small sample sizes are often dismissed as lacking sufficient relevancy.”
* “Some workarounds for handling the problem of CNN overfitting include (1) transfer learning, where the network is pretrained on a massive dataset and then finetuned for a specific problem and (2) data augmentation, where new samples are generated that are representative of the different classes. Some other methods that reduce overfitting include dropout, batch normalization and zero-shot/one-shot learning”
* [6]: image augmentation is kernel filters (blur and sharpen), color space tranforms, geometric tranformations, random erasing/cutting, image mixing. Caution must be taken to preserve labels. Rotation and translation to create new samples. Shifting to avoid positional bias in a set of images. (Translation often adds noise). [23] and [19] show that rotation performs better than other augmentation methods. [7] image mixing and data erasing

When shown to a small dataset size, network learns high variance which leads to a phenomenon called overfitting. Overfitting is when a model learns the training data too well, such that it is not able to generalize on unseen data. Different regularization methods have been deployed to tackle this problem. This includes (1) Data Augmentation, (2) Patch Extraction, (3) Pretraining. Other regularization methods such as Dropout (references in shorten), …, … have not been investigated in this study and refer to future work.

**Introduction**

melt ponds have a significant influence on the amount of sea ice melt [[*Perovich et al.*, 2002a](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0033); [*Tschudi et al.*, 2008](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0050)], on Earth's radiation balance [[*Maslanik et al.*, 2007](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0026); [*Perovich et al.*, 2007](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0035); [*Nicolaus et al.*, 2010](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0028)], and the potential loss of a multiyear ice coverage [[*Maslanik et al.*, 2007](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0026); [*Perovich et al.*, 2007](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0035); [*Nicolaus et al.*, 2010](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0028); [*Kwok and Untersteiner*, 2011](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0020); [*Serreze and Barry*, 2011](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0046)] (https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869)

* ‘A quantification of the overall distribution of melt ponds would be helpful to constrain the role of sea ice for the Arctic amplification and Earth's climate system [e.g., [*Holland et al.*, 2006](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0015); [*Eisenman and Wettlaufer*, 2009](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0007); [*Notz*, 2009](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0029); [*Serreze*, 2011](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0045); [*Serreze et al.*, 2011](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0047); [*Kurtz et al.*, 2011](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0019); [*Perovich et al.*, 2011a](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0038)]’
* Until now, statements about the melt pond distribution in the Arctic can only be made from the attempts to model melt ponds [*[Lüthje et al.](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869" \l "jgrc12461-bib-0023)*[, 2006](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869" \l "jgrc12461-bib-0023); [*Pederson et al.*, 2009](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0030); [*Scott and Feltham*, 2010](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0044); [*Skyllingstad et al.*, 2009](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0048); [*Flocco et al.*, 2010](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0011)]. A realistic presentation of melt pond fractions in the Arctic is only be possible with observations on a large scale over at least one melting period. Therefore, it is important to use remote sensing techniques that are applicable to detect the evolution of melt ponds. To survey melt ponds Arctic-wide, approaches regarding the use of satellite data have been developed by[*Markus et al.* [2003]](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0024); [*Tschudi et al.* [2008]](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0050); [*Rösel and Kaleschke* [2011]](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0040) and [*Rösel et al.* [2012]](https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2011JC007869#jgrc12461-bib-0041).

Unmixing algorithms based on multi-spectral optical data are a promising method for retrieving large-scale pond fraction but they have quite large uncertainties due to prevailing clouds and cannot derive detailed pond geometry (size and shape) which is important to understand the melt pond development. High-resolution satellites (e.g. SAR) provide more accurate all-weather ice-surface information (Yackel and Barber 2000; Kim 2013; Mäkynen 2014; Scharien 2014) but their narrow swath and inability to discriminate between melt ponds and leads enclosed within interconnected ice floes limits their ability to retrieve a basin-scale surface fraction. Landsat’s low temporal resolution and incapability of cloud penetration hampers its usage in obtaining a daily or weekly datasets of melt pond coverage (Markus 2003). <https://www.cambridge.org/core/journals/annals-of-glaciology/article/melt-pond-distribution-and-geometry-in-high-arctic-sea-ice-derived-from-aerial-investigations/8770BFD400443CB8704AE5B32D3577AE>

To retrieve melt pond fraction from image data, surface classes need to be extracted. This is the task of semantic segmentation. Recent work has mainly focused on detecting ponds in optical images that capture the amount of sunlight reflected by surface objects in the visible range. However, optical images are affected by clouds (<https://www.researchgate.net/publication/341944175_Machine_learning_approaches_to_retrieve_pan-Arctic_melt_ponds_from_visible_satellite_imagery>) and are not available at winter or night during polar night when no sunlight is available. Unaffected by these conditions are thermal infrared images that capture the thermal radiation emitted by objects, corresponding to their surface temperature emitted by the underlying object.

Melt Ponds extracted from IR images could be applied for seasonal prediction from temperature anomalies in winter more accurately as in (<https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/2022GL101493>)

* Measured brightness temperature is close to actual surface temperature, thus adding additional information about thermal properties of the surface structures observed.
* (Reference for sea ice surface temperature = brightness temperature)

**Background**

Spatially and temporally changing surface temperatures make automated segmentation of melt ponds from IR images challenging.

Prior algorithms for segmenting melt pond images have mainly focus on optical image data.

Thresholding / color based detection / pixel-intensity based segmentation:

A very simple method consists of setting a threshold value on single or multiple color channels and classifying accordingly. Some approaches use thresholding of single color channels (…) or combinations of channels (…). This is not applicable for IR images. (Thielke et al) prone to many errors, requires careful manual threshold choices that need to be adjusted for different environments and seasonal changes. Even within one flight, temperatures change for different surface features and even within one image, differences can occur (Figure 2). (changing temperatures due to changing atmospheric conditions, affected by flight).

<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2019JC015738> classified surface types with color-based segmentation and thresholds based on color channel combinations (= color-based segmentation)

<https://www.cambridge.org/core/journals/annals-of-glaciology/article/melt-pond-distribution-and-geometry-in-high-arctic-sea-ice-derived-from-aerial-investigations/8770BFD400443CB8704AE5B32D3577AE> , Perovich 2002, Inoue 2008, Lu 2010, 2011, Krumpen 2011): manually selecting red, green and blue RGB thresholds based on color distribution histograms of each image independently 🡪 time-consuming and does not work because of color-based

<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1002/2015JC011030> (**thresholding** and then consideration of neighbouring pixels intensities, high-resolution satellite)

<https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2000JD900275> (Tschudi 2001)

<https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/2022GL101493> (Linda; used thresholding for IR 🡪 exhaustive and prone to errors and small variations in surface temperatures)

Supervised approaches:

[https://www.researchgate.net/publication/341944175\_Machine\_learning\_approaches\_to\_retrieve\_pan-Arctic\_melt\_ponds\_from\_visible\_satellite\_imagery (2020](https://www.researchgate.net/publication/341944175_Machine_learning_approaches_to_retrieve_pan-Arctic_melt_ponds_from_visible_satellite_imagery%20(2020)) (used a neural network to perform pixel-wise classification, however, this cannot take larger features into account and is equally unsuitable as thresholding).

<https://zenodo.org/record/7548469> (PASTA ice Github Repo Niels Fuchs; uses pixel-wise RF classifier)

<https://tc.copernicus.org/articles/8/2163/2014/tc-8-2163-2014.pdf> (decision tree)

Other work proved successful with Edge-based segmentation.

<https://tc.copernicus.org/articles/12/1307/2018/> “use an image segmentation algorithm to divide the image into objects which are then classified with random forest machine learning” (code: Wright 2017). Image segmentation algorithm: Edge detection to find boundaries between surface types, watershed to build complete objects based on edge locations and intensity.

<https://www.researchgate.net/publication/272365874_Regional_melt-pond_fraction_and_albedo_of_thin_Arctic_first-year_drift_ice_in_late_summer> (boundary segmentation with Otsu (chooses threshold to minimize intra-class variance of black and white pixels), boundary tracing with Moore-Neighbor tracing algorithm, thresholding in red channel).

In IR images, boundaries are often not sharp enough to encounter for accurate edge detection / However, accurate edge detection requires clear boundaries of surface structures which is not given in IR images (see Figure).

The impracticality of existing approaches requires to develop a new method that can detect more complex features and takes shapes into account. In recent years, major improvements in image classification and segmentation could be achieved with CNN-based architectures, have been applied in the sea ice domain (<https://www.mdpi.com/2077-1312/8/10/770>;

CNN???

<https://tc.copernicus.org/articles/13/2421/2019/> (Neural Network to estimate snow depth on Arctic sea ice with microwave radiometry satellite data)

<https://ieeexplore.ieee.org/abstract/document/9443178> (Used an ensemble neural network segmentation approach for ship-based images 🡪 (very complex approach, not easy to determine effects of single components))

<https://ieeexplore.ieee.org/abstract/document/7122229> (Neural Network for pixel-based classification of SAR

meltpondnet

**Melt Pond optical properties**

<https://www.cambridge.org/core/journals/annals-of-glaciology/article/melt-pond-distribution-and-geometry-in-high-arctic-sea-ice-derived-from-aerial-investigations/8770BFD400443CB8704AE5B32D3577AE>

Generally <10m in size. (of order 10m² in area at melt onset (Perovich 2002)

The geometric appearance of melt ponds is relevant for detection.

Can vary in shape, ranging from circular or oval to irregular or elongated (influenced by factors such as the underlying topography), size and area (small ponds to large expanses covering significant portions of the ice surface). Some ice floes with lots, some without. Surface temperature (influenced by depth). Melt ponds can be interconnected, forming a network of channels and pools on the ice surface. Generally, they occur as warmer pools on the top of ice floes, although this cannot be universalized.

These characteristic properties are hoped to be detected by feature extractor of neural network.

**U-Net**

Two main parts: Feature extractor or backbone which takes an image as input and progressively reduces the feature space’s dimension, producing a highly nonlinear representation of the image. Second, the upsampling part / decoder, which utilizes features from the feature extractor and outputs a segmentation mask.

* Batch normalization added, regularization technique that normalizes set of activations in a layer and shown to reduce overfitting (reference). Substracting the batch mean from each activation and dividing by the batch standard deviation Sergey I, Christan S. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: ICML; 2015

**Training Data**

Neural Network architectures generally perform better

**Preprocessing**

**Patch Extraction**

**Augmentation**

**Pretraining**

The thesis is structured as following.

**--------------------------------------------------------------------------**

**Methods**

**Dataset**

Melt Pond size

For autoencoder: every 8th image

Place of collection (central arctic?)

* Retrieved from IUP server, as netCDF files
* Why ungridding was not suitable
* Brightness / atmospheric correction has been done before

**Annotation**

* Labelling process took a lot of time 🡪 several per image. Therefore, only 8 images were labelled in total.
* Training images have been selected to accord for diversity (different sizes, shapes and temperatures of features) and visibility to accord for accurate labelling
* Temperatures have been clipped to 273,276 which increases contrast
* To allow for finer labelling control, images have been upscaled to 2345,… and later downscaled (🡪 might have caused interpolation inbetween)
* …. Mask has been created
* Gimp as labelling tool for manual control/ high costumizability/fine-tuning 🡪 load images and masks, carefully fintetune outlines and fill with according class colour, with visual image as control.
* Labelme and AI segmentation tools didn’t yield good results (WHY)
* Commonly used AI driven tools like LabelMe didn’t lead to promising results in first experiments
* Checked and optimized labelling several times
* For model feeding, masks have been one hot encoded (WHY)

**Uncertainties in Training Data (maybe discussion section)**

Edge Melt ponds: It is discussed if edges count as melt ponds. Melt ponds at the edge of ice floes can occur when an ice flow breaks at a melt pond. Edge melt ponds have been labelled as melt ponds, mostly in use of optical images.

Melt Ponds visible in the optical range but not in the IR range: WHY. These ponds have not been labelled as melt ponds.

Melt ponds in optical images, only light temperature changes in IR 🡪 not labelled because hardly distinguishable (also not detected by edge detection algorithm)

Shapes of melt ponds in IR sometimes hard to identify because of blurred edges 🡪 adapt from optical

Darkblue melt ponds 🡪 no detection by algorithm, hard to define shape and ambiguous (sometimes it is one in optical, sometimes not)

Sometimes smaller ice floes at edges of bigger floes are not matching with real shape

Observation: Melt ponds can be blue (colder) or lighter (warmer)

Annotation has been revised

**Crossfold validation**

As due to the small dataset size, it is likely that test and train set are not drawn from the same distribution, which is important. This will be more obvious for larger patch\_sizes and non-offline augmentation, when the train and test set sizes are even smaller. Test set has very low diversity. To be able to overcome this bias, 10-crossfold validation has been performed on the final model. 10 has been chosen as parameter k as often done in literature. Note that, for more comparable approach, crossfold validation would have been needed for each evaluation step, however, this would have resulted in exploding computational costs.

(number of images per fold. Report results in terms of average accuracy and standard deviation among the 10 folds. For a given fold, we compute the accuracy for each class and then compute the average accuracy among the classes. This accuracy is used to compute the final average accuracy among the 10 folds.)

“We split the training dataset into five folds, each one using … for training and the fifth for validation. On every iteration, a model for every classifier was trained using the training data folds and validated on the validation one.

We split the dataset into 5 fold. During training, the first fold gets allocated as test dataset and others used for training while evaluating the model using the first fold as test set. In the next iteration, we use the second fold as test set and the other folds for training, and continue the process with all k folds. This way, we’ll get k scores corresponding to each of the k folds being used as test dataset.

Kfold splitting is done with random state, such that different experiments (when using the same dataset) will use the same splits.

**(Later) Ablation for VIS images (further: thesis\_structure2)**

Manually labelling allows for high costumisability (accuracy) but is very time-intensive. Small dataset leads to suboptimal results. An automated labelling approach could yield more training images in less amount of time.

* Optical images by AWI as .jpg files, matched to corresponding IR via timestamp. Optical images were taken every 4th second 🡪 every 4th IR image matches one optical image.

(FIGURE COLOR HISTOGRAM OPTICAL)

Optical images are available. DESCRIBE WHY MELT PONDS ARE VISIBLE IN OPTICAL RANGE.

* Final labelling method is based on manual evaluation
* Kmeans because fast and efficient (DESCRIBE KMEANS), no a priori knowledge required, only one parameter
* Used sklearn implementation because costum initialization possible
* Color distribution within the available images roughly the same (same lighting conditions) 🡪 kmeans suitable with same classes
* RGB and HSV did not yield better results than kmeans 🡪 grayscale less data and faster processing time
* Following pipeline was chosen due to experiments on a subset of 10 images
* To create reproducible results, 4 max have been selected as initial clusters
* 4 classes and merging more accurate results than 3 classes (tested different number of k on 10 randomly selected images; elbow method said k=7 but time-intensive and more difficult to merge afterwards; 4 led to good results
* Smaller errors in melt ponds could be postprocessed with area closing with high threshold.
* Select only images were labelling worked and finetune small misclassifiactions

**Matching**

* First experiments with Gimp (unified transformation), Hugin, Agisoft 🡪 no good results
* Pre-alignment with matlab: 5 control points; Lorenzos algorithm; mismatched edges 🡪 crop center

FIGURE OF MATCHING RESULTS

**Patch Extraction**

All images in size (640,480) have been center cropped to (480,480). This excludes some distortions that are more present on the image edges and makes sure that images are compatible with model requirements (WHY).

Single melt ponds cover only small parts of the images. Hence, not full image size can be suitable as training data. Smaller patches will lead to a larger training set size, but looses receptive field and thus context, making model unable to learn underlying structures. Patches should be large enough to cover melt pond shape, interconnections/channels and situation on ice floes. Not too big such that big scale structure changes are not included (such that changes in ice floe patterns etc).

(patch sizes also required because model constraints to be divisible by 32)

(discuss important optical properties of melt ponds 🡪 patch size)

The following statements are image-dependent, as floes vary a lot in size and shape, but roughly:

480: Covers full image resolution cropped, leads to very small dataset

256:

128:

64:

32: **small patches cover little structure**

To create least overlapping patches, sliding window approach has been used. 256 not included in 480 🡪 thus some overlap. Resulting number of patches are listed in Table.

(Table with dataset size; Figures with patch examples)

**Pretraining**

Imagenet was chosen as pretraining dataset for the following reasons: (1) large dataset with 1000s of images, (2) publicly available and easy to access as UNet backbone.

Imagenet is a popular benchmark dataset that contains … images (size: …) of .. . The data won … challenges, is available at …, divided into … classes

**Training Pipeline**

**(create Figure showing training pipeline)**

Train each state with 5 fold crossvalidation and report average iou +- standard deviation

* Mid-range CPU, 2 Kerne, 4 threads, Intel 6. Generation. “Calculations were made on a PC with 2-core Intel Core i7 CPU @3.1 GHz
* The models trained using Keras with TensorFlow as its backend. When kernels are initialized, we use the Glorot uniform distribution of weights.
* 56, 57, 58, 59 (dropout)
* We refer to … as the “baseline” model
* 64: Adam is best overall optimizer choice, 65: adaptive methods like Adam have a worse generalization than SGD. RS report best results of Adamax. (🡪 refer to RS paper again!)

For model implementation, data pipeline and augmentation example, used sm library.

Pre:

* Baseline arguments have been taken, patch size 256 (as this performed quite well on preliminary tests) and three different backbones tested: vgg19, resnet34 and inceptionv3
* Inceptionv3 has been best, although not much difference to resnet. Vgg resulted in unstable training. Results in appendix.
* Resnet chosen as baseline because inception net implementation not available for patches smaller than 75
* Baseline model with default parameters for all patch sizes (Pretrain=imagenet, batch size = 4 bzw. 2, optimizer = Adam, categorical crossentropy loss).
* Best performance (256) very small test set 🡪 try crossfold to see if makes difference (variance …); to save training time and resources only perform on this setting CAREFUL WHEN CROSSFOLD + OFFLINE AUGMENTATION 🡪 AUGMENTATION WILL GET INTO TEST SET
* Final augmentation methods were chosen by testing on one model and increase (start with none, geometrical, blurring, brightness…), were tested on fly because faster
* Mode 0 and 1 couldn’t improve without onfly augmentation, performed equally well (maybe train for more epochs in end)
* Mode 2 (adding brightness and contrast) resulted in worse performance and unstable training 🡪 brightness contrast changes were disregarded for mode 3 and mode 4
* Mode 3 (adding sharpening and blurring) resulted in worse performance 🡪 sharpening and blurring was disregarded for mode 4
* Training evaluation on mean iou score
* Training was successful and models converged
* Further experiments were performed on best performing patch size only to save training time and resources
* Pretraining method
* Hyperparameter optimization with weights and biases (random search)
  + Backbones (for imagenet pretraining)
  + Augmentation method: Offline or online
  + Batch Size: im\_size / fraction
  + Loss: ‘categorical cross entropy’, ‘focal dice loss’, ‘dice loss’
  + Optimizer: Adam, nadam, ...
  + Class weights while training yes or no
* ‘winner’ was chosen and again trained with crossfold validation to get a better estimation of mean final model performance, (and maybe with more epochs)
* IMPLEMENT PER CLASS IOU FOR FINAL MODEL
* Take winner model and train again with vis images
* Compare MPF predicted by final model with MPF of optical masks (only works for IR model)
* (test final model on gridded version)

**Results**

Evaluation on MeanIoU score and per class IoU score.

**Patch Size**

Show loss and IoU of different patch sizes baseline (decide if table or plots)

Show predictions on ten examples

**Pretraining**

Show loss and IoU of different pretraining methods, 256

**Hyperparameter Optimization results**

**VIS vs. IR**

**Further research**

* Discuss patterns that could be observed in terms of MPF (melt ponds are regarded smaller / larger…)
* More training data needed, more diverse training data (before mention that flight 9 was chosen and had diverse data, for the size of training set chosen). VIS only flight 9 because optical data was only available for flight 9 (and 16?)
* Investigate more on matching VIS with IR and take more elaborated VIS segmentation into account (discuss disadvantages of kmeans). As IR labelling very expensive (and still prone to error)
* Kmeans: works for data at hand because similar conditions, might not work for different light etc, unstable, large amounts of processed data had to be disregarded due to labelling errors. Worked for this smaller study but for yielding large amounts of training data (even from different flights) different methods that are better automizable should be regared.
* Kmeans assigns clusters disregardless of pixel connectivity and context
* Investigate different models and attention for UNet (research what went wrong and how could this be addressed), ensemble of backbones, semi-automated approaches
* Ungridded data (model might learn distortional shapes)
* Melt pond temperature does not necessarily correlate with real melt ponds, mention border cases (have been labelled as melt ponds)
* Evaluation on different days and in different areas

**Discussion**

**Patch Size**

Discuss patch size performance (too small limitations – to large less training data). Research on training size vs patch size.

**Pretraining**

Discuss pretraining performance

Then future research.

Flow-chart of image processing

**Data**

“IR images from the ATWAICE Campaign 2022, acquired on afternoon flight on 18.07. Each image is 640x480 pixels and the ground sampling distance is … m/pixel.

Each pixel corresponds to a specific temperature value. Warmer objects appear brighter, while cooler objects appear darker.

Ungridded data.

Helicopter-borne imaging TIR. Study area. VarioCam HD head 680 camera with a brightness temperature precision of 0.02K and accuracy of 1K (reference). Gridded surface temperatures at 1m resolution. Focus on data from 18 August that was mostly unaffected by atmospheric conditions (as has been done in <https://www.nature.com/articles/s41597-022-01461-9.pdf> ???)

(additionally optical during polar day, Lena AWI)

The surface classes captured in the infrared image have different spectral signatures. A IR camera has one spectral band at wavelengths of around …nm. The digital number of a pixel (DN) contains part of the spectral signature of the different classes. In IR, this digital number encodes the brightness temperature and strongly correlates with the surface temperature of the respective class.

Dataset Selection

Within Flight 9, images have been selected according to having a lot of variation and making the model able to analyse a large variety of images. In future, even more Flight should be taken into account to account for a larger diversity in the dataset.

CLASS IMBALANCE (network will be biased towards the majority classes 🡪 oversampling?)

* Make loss decision due to class imbalance
* <https://www.cv-foundation.org//openaccess/content_cvpr_2016_workshops/w19/papers/Kampffmeyer_Semantic_Segmentation_of_CVPR_2016_paper.pdf>

**Annotation**

To encounter for better visibility during annotation, the images have been clipped to a temperature range of [273, 276] allowing for a higher contrast.

To help for Gimp, masks have been aligned with the images. (describe Sobel Filter…) High gradient values correspond to abrupt changes in pixel intensity, which are likely boundaries between surface types. (no further noise detection and filtering of weaker edges). However, melt ponds in IR are often not clearly distinguishable by boundary, which led to further manual processing of several hours per image, to account for accurate training masks.

**Patch Extraction**

**Augmentation**

KOMBINIEREN MIT ORIGINALEN DATA DAMIT GRÖßERER DATENSATZ

* To make model more robust and prevent overfitting 🡪 make dataset more diverse
* Increase dataset size, good method to increase without manual labelling required (images and masks will be augmented accordingly)

Methods used:

* Noise injection (Gaussian noise)
* Flipping
* Rotation
* Blurring and sharpening
* Brightness contrast

Methods not used:

* Color augmentations (saturation)
* Multiple image augmentations (mixing strategies have been disregarded; mixup results in unrealistic outputs and label ambiguity; cutmix will distort object boundaries)

Images were taken of an overhead perspective and therefore require no specific orientation. Flipping, random rotation could be applied. Cropping and zooming were disregarded as augmentation techniques as patch extraction should be observed and this would have distorted effects achieved by patching.

Random brightness and contrast changes were applied. No absolute information from temperature values.

More advanced Multi image augmentation techniques that combine multiple images by blending or cropping like mixup or cutmix (successfully applied in recent studies) have been disregarded because this would have distorted the closed and unique shape of melt ponds.

Sharpening and blurring have been applied as these could imitate atmospheric effects of water vapour or changing camera quality.

(Robustness measures for varying quality) Image blur imitates snow, rain, water on camera lens; brightness decrease imitates changes in surface temperature due to different points in space or time; water vapour that results in blurring the IR image)

“During training, random image augmentations were performed on the images: Random flipping, rotation, zoom, brightness, contrast, (hue and saturation 🡪 optical). As temperatures are not consistent among the surface types, brightness change was still selected, imitating an even higher variability in environmental conditions. 🡪 diversify and enlarge training dataset”.

Cropping and zooming were disregarded as augmentation methods because it would have distorted the effect of different patch sizes, which was under study.

‘Albumentations is a fast and flexible Python tool for image augmentation. It is widely used in ML competitions, industry and research to improve the performance of DL”

**Model Training**

Adam Optimizer: <https://arxiv.org/abs/1412.6980>

**Model Evaluation**

Model Evaluation is a important method to estimate the models performance and be able to compare different hyperparameter settings. Train test splits allow to detect overfitting.

Model Evaluation has been done using a train test split of 70 / 15 / 15 (random selection; splits only calculated once and reused for all experiments to allow for comparibility. However, note that it results in different training and testing sets for the different patch sizes). As the training set is very small, for some patch sizes, this results in a test set that is only one image and therefore likely to be highly biased (reference). Hyperparameter Optimization is dependent on the evaluation metric. K-cross validation allows to overcome this issue but results in heavy training. Therefore, k-cross validation has been performed on a default hyperparameter setting, the results compared to a simple train test split.

(k was chosen as five because it achieves a balance between the number of images in the validation sets and number of folds for cross-validation). 🡪 Divide into six parts of nearly equal size, use first five for cross-validation and set sixth fold aside as test set.

<https://ieeexplore.ieee.org/abstract/document/9443178> (histogram to make sure that class distribution among folds is equal).

**Pretraining**

A common method to pretrain U-Net is exchanging the encoder with a pretrained classification backbone as Resnet34 or VGG16.

Commonly, networks are pretrained on Imagenet. This has been given successful results even for sea ice domain (Petersen et al 2009), however,

Note that the results are not directly comparable between plain U-Net, pretrained with ResNet and pretrained autoencoder, as the ResNet is used as backbone.

(Training: The pretrained backbone was frozen, only the upsampling part was trained). For UNet, Keras implementation was used and adjusted. Autoencoder was implemented with inspiration from…

**Optical Training Data**

<https://www.sciencedirect.com/science/article/pii/S0165232X09000032?casa_token=CYND2U8oJU0AAAAA:6uob_5DIJWHREBWz8ahxK8E66WT1tM6QzkleGjuZjCISKrkklNT6760zYytnieMroBIcTE9E_XUz> used kmeans to classify into different ice types (successful given appropriate image conditions and environment (e.g. illumination, visual clarity, snow cover).

Important: are the aerial images preprocessed / normalized (think so)

**Hyperparameter Optimization**

**Results**

Even if trained with a very small dataset, UNet could achieve quite good result. This is promising as when dataset size is increased, this could further be improved and overfitting reduced. (Going large-scale is more difficult: Images are more diverse, corresponding to various acquisition conditions and times. Further augmentation could account for some of these diversions but more data must be labelled for diverse settings).

**Evaluation Metrics**

5-fold cross-validation was used as evaluation scheme (to evaluate the model on each of the five validation sets, the neural network was trained on the remaining four validation sets).

Quantitative Performance indicators:

MeanIOU

( Accuracy (biased towards most frequent class) )

F1score

IOUc (for particular class)

Mean IOU has been selected as primary metric to make strict decisions possible, as standard in image segmentation literature

**Quantitative Results**

**Baseline**

**Ablation on patch size**

**Ablation on pretraining approach**

**Ablation on training dataset**

**Time Complexity**

**( Labeling Consistency )**

**Qualitative Results**

Qualitative Results contained visual inspection of segmentation results.

**Melt Pond Fraction**

Knowing Melt Pond Fractions sufficient for crudely estimating regional averages of albedo and surface ablation.

„Melt Pond fraction is the ponded area relative to the sea ice area (Webster 2015). We define MPF as the ponded percentage of the sea ice area:

MPF = ( MP / MP + ICE ) x 100, as has been done in (<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2019JC015738>).

To allow for another evaluation metric, MP fraction has been calculated for RGB and predicted IR images and compared to MPF calculated with … method.

**Discussion**

<https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2019JC015738> (discuss factors that lead to misclassification in RGB images color-based)

Difficulties of manual segmentation: Although done very carefully, some melt ponds could not be detected.

Problem with unsupervised kmeans segmentation: Produce inconsistent, non-intercomparable results (fixed k, even if not all features are present).

Distinction melt ponds and open leads. “Submerged ice (resulting from lateral melting) presented at some ice floe edges are classified as melt ponds and melt holes (melt ponds that penetrated through the underlying sea ice cover) are classified as leads, partly because of similar optical and physical properties between submerged ice and pod, and between hole and lead (Inuoe 2008) <https://www.cambridge.org/core/journals/annals-of-glaciology/article/melt-pond-distribution-and-geometry-in-high-arctic-sea-ice-derived-from-aerial-investigations/8770BFD400443CB8704AE5B32D3577AE> . Fraction of these features are very small compared with the melt ponds and image coverage, resultant errors with the mean value of pond statistics for one image are likely to be small (Perovich 2002) 🡪 (but might be too large for training data).

Notice that the aerial photograph is highly distorted, and hence, the sizes of melt ponds in the aerial photograph cannot be directly compared with those in the airborne SAR image acquired in this study.

IR data was fed in ungridded. This was encountered by using the image centers that generally are less distorted. However, this could effect model performance by learning wrong shapes.

Necessary to label more images, also from different flights and possibly different seasons.

Future work: Work more on alignment of optical an IR images, improve classification method of optical (apply discriminant analysis to find highly discriminative features while reducing the dimensionality of the feature space).

Implement different loss function, further tune hyperparameters, extract different number of patches from images.

Different architectures such as PSP-Net (uses pyramid pooling module to aggregate multiscale context information in different subregions). DeepLabV3+ combines encoder-decoder structure with SPP to better capture multiscale contextual information and produce sharper object boundaries by gradually recovering spatial information. However, they are more complex and require more training time.

**Conclusion**

The objective of this work was to develop a deep learning-based approach to segment helicopter-borne IR image data containing three classes (ocean, sea ice, melt pond) that can later be used as validation data for satellite imagery.

The results can be summarized the following:

* … classes and … could be detected well, other poorly.
* Pretraining with … could outperform pretraining with IR images (on average and in per-class IOU).

To be applicable in future and derive correct MPF, data needs to be gridded.

“Further applications include calculation of partial and total concentrations, melt-pond feature size assessment

**References**

[1] Flocco, D., Schroeder, D., Feltham, D. L., and Hunke, E. C. (2012), Impact of melt ponds on Arctic sea ice simulations from 1990 to 2007, *J. Geophys. Res.*, 117, C09032, doi:[10.1029/2012JC008195](https://doi.org/10.1029/2012JC008195).

[2] Fetterer, F., and Untersteiner, N. (1998), Observations of melt ponds on Arctic sea ice, *J. Geophys. Res.*, 103( C11), 24821– 24835, doi:[10.1029/98JC02034](https://doi.org/10.1029/98JC02034).

[3] Rantanen, M., Karpechko, A.Y., Lipponen, A. *et al.* The Arctic has warmed nearly four times faster than the globe since 1979. *Commun Earth Environ* **3**, 168 (2022). <https://doi.org/10.1038/s43247-022-00498-3>

[4] Wright, N. C., & Polashenski, C. M. (2020). How machine learning and high-resolution imagery can improve melt pond retrieval from MODIS over current spectral unmixing techniques. *Journal of Geophysical Research: Oceans*, 125, e2019JC015569. <https://doi.org/10.1029/2019JC015569>

IOU standard in literature:

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

Z. Zhang, "Exfuse: Enhancing feature fusion for semantic segmentation", Proc. Eur. Conf. Comput. Vis., vol. 2018, pp. 269-284.

L.-C. Chen et al., "Searching for efficient multi-scale architectures for dense image prediction", arXiv:1809.04184, Sep. 2018, [online] Available: <https://arxiv.org/abs/1809.04184>.

H. Zhang, H. Zhang, C. Wang and J. Xie, Co-Occurrent Features in Semantic Segmentation, Sep. 2020, [online] Available: <http://hangzh.com/>.

X. Zhang et al., "DCNAS: Densely connected neural architecture search for semantic image segmentation", arXiv:2003.11883, Mar. 2020, [online] Available: http://arxiv.org/abs/2003.11883.