Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

Azamat Kaibaldiyev

# Self-supervised multimodal representations for marine robotics

Master's thesis in Marine Technology
Supervisor: Oscar Pizarro
Co-supervisor: Ricard Marxer
June 2023

NTNU
Norwegian University of
Science and Technology

Azamat Kaibaldiyev

# Self-supervised multimodal representations for marine robotics

Master's thesis in Marine Technology
Supervisor: Oscar Pizarro
Co-supervisor: Ricard Marxer
June 2023

Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

NTNU
Norwegian University of
Science and Technology

# Abstract

This thesis work explores multimodal learning techniques for habitat classification using remotely sensed and visual data. Autonomous Underwater Vehicles (AUVs) play a vital role in marine scientific surveys, providing efficient data collection and observations over marine ecosystems. Benthic habitat mapping, which involves classifying seabed sites into different habitat categories, is a key objective in marine ecology. AUVs capture visual imagery of the seabed, while multibeam sonars collect bathymetry data. By correlating visual imagery with features from the bathymetry data, reliable habitat classification models can be developed. This study investigates self-supervised learning approaches, particularly contrastive learning, to enable robust classification and image-content prediction. Results show that contrastive learning on bathymetry data achieves test accuracy rates of approximately 59% and 63% for patch sizes 16x16 and 32x32, respectively. In contrast, visual imagery achieves over 86% accuracy. Multimodal learning, combining visual images with bathymetry patches, yields accuracies of about 71% and 72% for different patch sizes. Separate networks with shared loss achieve accuracies of over 71%. This work demonstrates the feasibility and effectiveness of multimodal learning techniques in habitat classification, leveraging the strengths of both visual and bathymetry data. Future work involves exploring additional self-supervised multimodal learning approaches to improve underwater data analysis.

# Acknowledgements

I would like to take this opportunity to thank all those who helped me to finish my master's thesis.

First of all, I would like to express my sincere gratitude to my supervisors, Oscar Pizarro and Ricard Marxer. Their support, wisdom, and guidance during my master thesis journey were crucial for me and also fostered my development as a researcher.

To my family, father, brother Darmen and sister Aigerim, I would like to extend my deepest appreciation for your love, understanding and support. Their belief in my abilities and their encouragement have been my driving force.

And finally, I just want to express my heartfelt gratitude to my girlfriend Diana. From the beginning of my thesis journey her words of support and love has inspired and motivated me.

# Table of Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Motivation

Nowadays Autonomous Underwater Vehicles (AUVs) can be advantageously integrated in marine scientific survey tasks. They can perform efficient data collection in areas with difficult accessibility, by providing longer operation time and excellent data extraction capacity, thus facilitating the observations over marine ecosystems. One of the main objectives in marine ecology is benthic habitat mapping, which is the task of performing classification of sea-bed sites into different habitat categories e.g coral reefs, sand, seagrasses etc (Williams et al. 2010). AUVs can be deployed to perform benthic imaging where they extract numerous amount of visual imagery by the camera with corresponding geo-reference on planned paths. This visual imagery provides high spatial and angular resolution data that is easy for humans to interpret and is further used for habitat classification and benthic research of the area under investigation. However, the problem is that deploying AUVs to obtain the whole imagery of the sea-bed surface is not feasible due to time and cost constraints. The marine seafloors of interest can cover thousands of square kilometers, which may require substantial amount of human and machine resources. AUV's image sensor footprint is rather typically limited to ranges of a few meters at most, covering area of between 1 and 10 meters squared. On the other hand, multi-beam sonars are of great use, since they can obtain seabed bathymetry data of vastly extending sizes in time-efficient and low-cost manner, yet data is comparitevely of low resolution. Also, this acoustic imagery has much longer ranges and is not affected significantly by water turbidity but can be harder to interpret. Some underwater robots operating near the seafloor often carry multiple sensing modalities to complement the strengths and overcome the limitations of each modality. For example, visual and acoustic imagery are present together, when robot has both sonar and visual camera equipment.
Benthic habitats seem to have correlation to the their underlying bathymetry. Therefore, seabed terrain data can be utilized to identify habitat types of the corresponding regions by the means of machine learning predictive models. Specifically, this can be achieved by correlating imagery data which is considered to be ground-truth for habitat class and features extracted from the seabed bathymetry data. As a result, highly detailed maps of habitat types that spans over regions of huge sizes could be produced. Thus, there is a need to build reliable and robust models to precisely learn correlations between the scarce visual imagery and features from acoustic data of seabed terrain. This thesis work will investigate machine learning approaches in particular self-supervised learning with multiple modalities to enable robust forms of classification and image-content prediction.

## 1.2 Objectives and scope

This thesis aims to:

- review relevant literature in multimodal perception and its overall underwater applications
- define a specific scenario for multimodal perception underwater and explore self-supervised contrastive learning approach to solve it
- demonstrate and evaluate potential approaches using synthetic data, existing real-world data, and/or experimental data

This thesis aims to address the scarcity of ground truth data in underwater datasets by initially focusing on available remote sensing data of the Earth's land surface. The research scope encompasses the development and testing of a proposed model using digital elevation models of terrestrial terrain and satellite imagery with labeled information. The reason for incorporating terrestrial data lies in its advantage of having available ground truth everywhere, which contrasts with the limited availability of ground truth in underwater datasets. The reason and importance of using terrestrial data is its advantage of having a groundtruth everywhere compared to underwater dataset. This initial stage serves as a foundational step for further development and evaluation of the model using underwater data, specifically datasets of bathymetry and underwater visual images. The central concept revolves around the application of self-supervised learning techniques to extract latent space representations from both the Digital Elevation Model and bathymetry patches. These representations will then be utilized for downstream tasks such as the classification of terrestrial land surfaces and underwater habitats, respectively. By employing this approach, the thesis aims to contribute to the advancement of knowledge and methodologies in analyzing remote sensing data for underwater environments.

## 1.3 Outline of report

This thesis work is structured as follows:

**Chapter 2** focuses on background and theory, discusses bathymetry, deep learning, unsupervised feature learning, self-supervised learning and contrastive learning.

**Chapter 3** is dedicated to literature review, where different methods utilizing remote sensing data are explored.

**Chapter 4** describes approach, datasets and implementation used in this paper.

**Chapter 5** shows obtained results.

**Chapter 6** discusses on the results and future improvements.

**Chapter 7** concludes the master thesis and offers future work directions.

# 2 Background and Theory

This section gives overview on bathymetry, visual images, benthic habitat classification, unsupervised feature learning and deep learning. Section 2.1 provides an explanation of bathymetry. Section 2.2 presents deep learning models. Section 2.3 discusses unsupervised feature learning techniques. Section 2.4 focuses on the exploration and explanation of self-supervised learning methods.

## 2.1 Bathymetry

Topographic maps shows the lay of land terrain, where the variations of depth are depicted color and contour lines. Bathymetry is similar to topographic maps, but represents submarine topography, which refers to the depth of the oceans with respect to the sea level. It also describes the depth as well as shapes of the underwater seabed. The data of bathymetry is obtained by using shipborn sonar such as multi-beam echosounder or sidescan sonar. The transmitter and receiver are mounted underneath the ship, from where transmitter sends series of acoustic pulses, which are reflected and then received by the sonar. The time between transmission and detection of pulses is used to compute the range to the ocean floor. The obtained data is then processed and resulting digital elevation map is created, which is two-dimensional matrix containing ocean depths values.

## 2.2 Deep Learning models

Deep learning is a machine learning method based of neural networks with representation learning. It uses multiple layers to gradually obtain high level representations of different scale and complexity from the given input. There are different architectures such as deep neural networks, deep belief networks, recurerent neural networks, feedforward neural networks, and convolutional neural networks. They are widely used in the areas of object recognition, classification, speech recognition, natural language processing etc. In this section, feedforward neural networks and convolutional neural networks will be introduced.

### 2.2.1 Feedforward Neural Networks

A Feedforward Neural network is the simplest architecture of neural networks as it processes information only in one direction and doesn't propagate backwards. Its simplest type is a single layer perceptron. Input vectors which pass through the model are multiplied by the weight vectors. The results are then summed to get the final weighted input values. If the sum is above certain threshold the activation function converts it to 1, otherwise to -1. These activation function are called linear threshold units. Furthermore, instead of just step function, single layer networks can be used to calculate continuous output. One of the examples is logistic function which turns network into logistic regerssion model:

$$f(x) = \frac{1}{1 + e^{-x}}$$

Multi-layer type architectures consist of multiple layers of hidden units. Each layer's activation function is usually a linear mapping, followed by nonlinear function such as sigmoid. The model, given an input x, calculates the output value and then is trained to minimize the error between its own output y and ground truth $y_true$. It uses gradient descent optimization technique, a backpropagation procedure, where the gradient of the error is computed with respect to the every layer. The common problem is that backpropagation is senstiive to vanishsing gradients, where the gradient of the error gets comparatively tiny to the lower layer parameters.

### 2.2.2 Convolutional Neural Networks

Convolutional neural networks are the most common types used for analyzing visual imagery, classification and computer vision tasks. They provide callablescalable approach for image classification and object recognition problems, but can be computationally demanding. CNNs are shift and space invariant, as they posses shared weights of convolution filters which are applied upon input features and provide feature maps. There are three mainntypes of layers in CNN:

- Convolutional layer
- Pooling layer
- Fully-connected layer

The convolutional layer is the first layer of the network, which can be followed by several pooling and the same convolutional layers. The final layer is the fully-connected layer. With each additional layers the complexity of CNN grows, where initial layers describe more simpler features e.g edges, and subsequent ones focus on larger features such as shapes etc. The most amount of computations happens in convolutional layers. It involves the process of convolution where an input, for example 3D RGB image with withd, heigh and depth dimension, are exposed to filters know as kernels which acts as feauture detectors. The filer is applied on small parts of an image with dot product multiplacation and moved across so that it covers the whole image. Each output value froms the final result which is a feature map.
Pooling layers are utilized to downsample the input by reducing its parameters. They use similar kernels as convolutional layers, but uses aggregation function instead of weights. Two common types of pooling are max pooling, where the filter selects the maximum value pixel, and avaerage poolin, where filter calculates the average valuein the receptive field.
Fully connected layers is used to perform the classification task based on features obtained from the other layers. The activation function of fully connected layers is usually a softmax activation function, which outputs probability ranging between 0 and 1.

## 2.3 Unsupervised Feature Learning models

In this part, unsupervised feature learning and their models, namely autoencoders, are described.

### 2.3.1 Overview

Supervised Learning models need huge amount of data in the form of input x and their corresponding labels y. On the other hand, the easily obtained and plentiful amount of unlabeled data can be handled by Unsupervised Learning which is trained without involving ground truth labels y. In addition to be able to learn better features than that of hand-crafted ones, they also can be integrated with the deep networks to build more powerful learning models. Here, the most common and popular type of Unsupervised Learning models, an Autoencoder and its variations will be briefly introduced.

### 2.3.2 Autoencoders

Autoencoder is an unsupervised learning technique used for the goal of representation learning. Neural network architecture of autoencoder produces compressed knowledge representation of the original input and further uses it to reconstruct the same input. Unlabeled dataset can be regarded as a supervised learning problem with objective to output $\hat{x}$, a reconstructed input x. The network is trained by minimizing the reconstruction error, which is $L(x,\hat{x})$, which computes the difference between the given input and generated output. The latent space generation is crucial for training and network design, since without it the network would just learn to memorize the input values by transferring values through the network. The latent space constraints the size of input information

that can pass through the network, thus forcing to learn compressed version of the original data. Autoencoders architecture consists of 3 parts:

1. Encoder part, which compresses high-dimensional input data into an encoded lower dimension that is usually smaller by several orders of magnitude.
2. Bottleneck part, which stores the compressed representation of the input information
3. Decoder part, which uses bottleneck module to decode the knowledge representation and reconstruct the original data. The result is compared to the original input.

The overall architecture is given in figure 1.

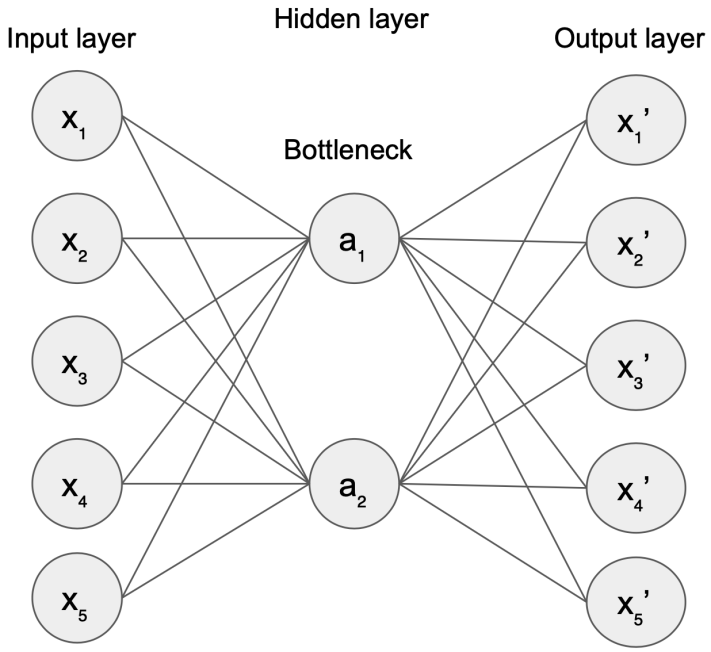



Figure 1: Overall architecture of an autoencoder

### 2.3.3 Denoising Autoencoders

Autoencoders that have deeper hidden layers have the risk of learning the identity function, which means that output will be directly equal to the input, making it unreliable. Here comes the idea of denoising autoencoders, which are extended version of the simple autoencoders. During training, a stochastic corruption is applied to the input x as a wat of regularising it. The corrupted input is then passed to the model, and the result is compared to the original uncorrupted input. This way denoising autoencoder will learn to rebuild the original input from noisy input by learning how to remove the noise to get the clean input and thus, become robust.
The process of applying noise is stochastic, which means each training input will be corrupted differently for each training iteration. Once the model is sufficiently trained with corrupted data, the latent space representation is extracted by inputting clean data to the model.

### 2.3.4 Variational Autoencoders

Basic autoencoders describe the features of the input as a value.On the other hand, Variational autoencoders(VAEs) describe the features of the input in latent space in a statistic manner, as a combination of mean latent vectors and standard deviation. Their encoder output a probability

distribution instead of single value. KL-divergence is applied in VAEs as a loss function, which tries to minimize the difference between original input distribution and supposed one. The loss consist of two terms, one of which is reconstruction error and the other one is Kl-divergence:

$$Loss = L(x, \hat{x}) + \sum_{j} KL(q_j(z|x)||p(z))$$

## 2.4 Self-supervised Learning

The problem of data annotation has drawn a lot of attention in the machine learning world. Unsupervised learning , semisupervised learning, weakly supervised learning , and meta-learning are only a few of the alternatives to traditional supervised learning that have been researched. Self-Supervised learning recently attracted a lot of interest in the field of computer vision and made substantial advancements toward the elimination of human supervision. Indeed, SSL methods already outperform supervised pretraining on many tasks by extracting representative features from unlabeled data (Goyal et al. 2021).
For the technique of self-supervison, a significant amount of unlabeled data is used to train a model which is usually a convolutional neural network for example, ResNet (He et al. 2016) or Vision Transformers (Dosovitskiy et al. 2020), by optimizing this objective without needing any manual annotation. The model f acquires the capability to capture high-level representations of the input data with a well crafted self-supervised task. After that, the trained model can be used to supervised downstream tasks for practical purposes.
The most prevalent and popular methods for creating such self-supervision often make use of three different kinds of goals: Rebuilding the provided input x, $f(x) \rightarrow x$; predicting a self-generated label, typically derived contextually and from augmenting the data (for example, estimating the order of splitted images); and comparing the semantic similarity of the inputs $x_1$ and $x_2$, where the compressed representations of two augmented views of a given image have to appear the same). The pre-trained model f could be used for downstream tasks assuming self-supervised training was effective. Models that are pre-trained using self-supervision, as opposed to supervised pretraining, have the ability to use more generalized representations and offer a technique to get around the drawbacks of supervised learning. Three benefits in particular exist for self-supervised pre-training: It does three things: (1) it eliminates the need for human annotation during pre-training; (2) it enables good performance on downstream tasks with a minimal amount of labeled examples; and (3) obtaining unlabeled data from the target application can ensure a small domain gap between pre-training and downstream datasets (Y. Wang et al. 2022). Self-supervised learning encompasses a diverse set of methods that can be classified into three distinct categories: predictive, generative, and contrastive methods . Generative methods within self-supervised learning focus on the reconstruction or generation of input data. Techniques such as Autoencoders and Generative Adversarial Networks (GANs) are employed to model the underlying data distribution and generate realistic samples. Predictive methods, on the other hand, aim to learn to predict self-generated labels or specific properties of the data. Pretext tasks are designed to provide supervisory signals for the model, allowing it to capture temporal, spatial, or spectral contexts and make accurate predictions. Contrastive methods tackle self-supervised learning by maximizing the similarity between semantically identical instances while minimizing the similarity between different instances. Negative sampling, clustering, knowledge distillation, and redundancy reduction techniques are commonly employed within this category(Y. Wang et al. 2022). By leveraging these three category methods, self-supervised learning approaches enable the acquisition of meaningful representations from unlabeled data, paving the way for a wide range of downstream tasks and applications.

### 2.4.1 Contrastive Learning

Contrastive representation learning essentially entails comparison-based learning. Contrastive learning focuses on learning representations by comparing various input samples, which is a major distinction between discriminative models that map data to labels or generative models that reconstruct input samples. Contrastive learning compares numerous samples, including positive pairings

of similar inputs and negative pairs of dissimilar inputs, as opposed to learning from individual data samples one at a time.
Contrastive learning techniques only call for defining the similarity distribution to sample a positive input $x^+$ from $p^+(* \mid x)$ and a data distribution for a negative input $x^-$ from $p^-(* \mid x)$ in relation to an input sample x, as opposed to supervised methods that demand a human annotation y for each input sample x. The main goal of contrastive learning is to make sure that comparable samples are represented near together in the embedding space while dissimilar samples are represented farther apart. To do this, positive and negative pairs of samples are contrasted, with the representations of the positive pairings being drawn closer together and the negative pairs being pushed farther apart.
Contrastive learning approaches are used in the self-supervised scenario to create a discriminative model based on several input pairs that share a concept of similarity rather than creating a pseudo-label from the pretext task. This notion of similarity can be ascertained from the data itself, as with other self-supervised tasks, overcoming a constraint in supervised learning where only a limited number of labeled pairings are available. Contrastive approaches are significantly easier to use because they do not require model architecture alterations between training and fine-tuning on subsequent tasks, unlike other self-supervised methods (Le-Khac et al. 2020). The general strucuture of conttrastive learning is shown in figure 2

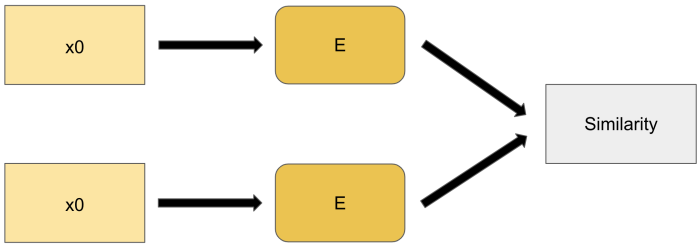



Figure 2: General structure of contrastive Self-Supervied Learning. E refers to the encoder, the loss is calculated in the representation space.

Source: Y. Wang et al. 2022

# 3 Literature review

A lot of existing deep learning methods apply backbone networks which are trained on huge datasets and fine-tuned for special problem. For remote sensing, shortage of comparably big annotated datasets makes it difficult to use similar approach. In (Heidler et al. 2021), the authors developed a self-supervised approach for pre-training deep neural networks without the need of annotated data by utilizing the relation between audio recordings linked to geographical location and remote sensing imagery. They also presented new dataset called 'SoundingEarth' which contains pairwise sets of audio and aerial imagery belonging to the same location all over the world. It consisted of 50,545 image-audio pairs, where the overall length of audio constituted over 3500 hours of sounds. Geo-tagged audio were collected using Radio Aporee ::: Maps project and consequently 1024x1024 pixels image corresponding to the same exact location was obtained from Google Earth with spatial resolution of 0.2m per pixel. This dataset was used to pre-train ResNet model to create new embedding space from the data from visual and audio modalities. A CNN network was chosen for both of these modalities to train and learn the common features between pairwise sets and perform the projection where embeddings of respective pairs are close, and if dissimilar pairs are apart. Rather than using conventional representation learning where a few sample embeddings are compared, they instead employ the idea of contrastive learning technique to put all pairings in a training batch and integrate it with triplet loss, and consequently get batch triplet loss. Given pairs of embeddings from both modalities (a,v), their pairwise distances are computed to construct the matrix D(a, v) and the aim becomes to minimize its diagonal entries, at the same time retaining others above some bound. During evalutaion process, this model outperformed other competing methods on a various benchmark datasets. They included aerial image classification, which is scene categorization into particular classes; audiovisual scene classification, ; aerial image segementation and cross-modal retrieval, where given an input image, the corresponding audio sample is predicted by extracting the nearest one from shared embedding space.

When the real-world annotated data is deficient and expensive to obtain, the other way is to train model on synthetic data like synthetic images. But it may lead to poor results because of notable difference between real and synthetic image distributions. In (Shrivastava et al. 2017), the authors focus on reducing this difference by introducing Simulated+Unsupervised learning approach which refines images from simulator's output and makes them more realistic by utilizing unlabeled real data. Their method which is named SimGAN is based on adversarial network similar to Generative adversarial netwroks (GANs), but instead of random vectors, inputs are synthetic images. The refiner network is fed with synthetic images from blackbox simulator as input, and trained with adversarial loss so that the output images are indiscernible from real images using a discriminative network. In addition to adversarial loss, annotations of synthetic data are retained with the help of self-regularization loss that impose penalty if there is a big difference between input and output images. In order to cope with drifting and artifacts, they constrain receptive field of discriminator to local regions and to improve its stability, the discriminator is updated with a history of refined images than only the current ones. For the quantitative evaluation, to test the visual quality of the refined images, visual Turing test was used, where subjects were asked to classify between real and refined samples. The results showed that they weren't able to correctly identify real ones from refined images.

The diversification of remote sensing platforms made it possible to obtain imagery from various sources like phone cameras, satellites and drones. Especially, unmanned aerial vehicle (UAV) enabled the application of image geo-localization to become a popular research topic. In (Liang et al. 2021), authors investigated cross-view geo-localization, where images taken from the satellite were matched with images taken from UAV to detect the same location. They propose the SNSnet model computes location distribution of feature vectors, where they use of siamese neural network fused attention mechanism and NetVlad: local features obtained from pre-trained resnet-50 model are inputted to spatial attention module and adjusted by VLAD vectors from NetVLAD, sent to classifier and their cross entropy loss is calculated. The local feature extraction, the first module of the model, is composed of pre-trained resnet-50 and a spatial attention module for increasing retrieval accuracy, enhancing important features and decreasing the weight of unrelated ones. The next module is the feature aggregation, where NetVLAD utilizes Vector of Locally Aggregated Descriptors (VlAD) encoding method to represent global features with the help of aggregated local

features for recognition of outdoor location. For the training and testing university-1652 dataset was used, where every location has a sateillite vertical view, drone oblique view and ground street view, the latter one supplemented by Google map street view images. For the evaluation part, the model achieved adequate results, where it was shown that while resnet-50 radicalized global aspects in the image, the author's approach accentuated on contextual information around the geo-target.

A lot of current semantic segmentation approaches based on deep-learning for remote sensing images need substantial amount of annotated data for training, which is expensive and laborious task. Self-supervised representation learning solves this issue, but focus on only one level features which presents negative influence on learning. In (Li et al. 2021), the authors tries to solve this by introducing a self-supervised multitask representation learning, in which triplet siamese network is used for high-level and low-level features learning for capturing effective visual representations. They build three different pretext tasks such as image inpainting to enable network to learn low-level representations, augmentation transform prediction (ATP) and contrastive learning to enable network to learn high-level representations. The inpainting branch restores occluded area from inputted random image, ATP predicts transformation of the input image and contrastive learning checks for pairwise similarity. Each one of them has its loss function, which are combined together and called multitask loss function. For the pretraining part, DIOR, DOTA and Levir, well-known remote sensing datasets were used, and LevirCS (cloud/snow detection), Potsdam and Vaihingen were used for semantic segmentation evaluation. For the evaluation part, four initialization methods were used: random initialization, image-net pretraining, self-supervised pretraining, and combination of the last two .Their proposed method showed better performance and results compared to other existing state-of-the-art self-supervised representation methods like NPID, MoCo and MoCov2, particularly with the availability of small amount of training data. However, with the larger amount of data available, the advantages of proposed method reduces.

Autonomous Underwater Vehicles are being extensively used for collecting huge amount of data for marine science research. One of the use cases is extracting close seafloor images to supplement bathymetric data collected from ships. However, as two types of data differs in scale, obtaining visual imagery to cover the whole bathymetric data map takes a lot of time and effort. In (Shields et al. 2020), authors present probabilistic habitat model that map remotely sensed data to the corresponding habitat class. They use Bayesian neural networks with probabilistic predictions for habitat modeling. First, habitat labels are assigned to images by ScSPM feature extraction and GMC clustering methods. Convolutional autoencoder is used for bathymetric data for obtaining bathymetric latent space, which then inputted to BNN probabilistic latent model. It estimates the uncertainty associated uncertainty, which then used to improve the model.

In (Castillo-Navarro et al. 2022), a new large-scale dataset called MiniFrance for semi-supervised semantic segmentation is presented. It contains 2000 high-res aerial images of various landscapes, fields, forests, urban and countryside scenes from different regions in France. Authors also introduce semi-supervised deep architectures and auxiliary losses such as the relaxed l-means loss for unsupervised semantic segmentation task. The first one is BerundaNet which is based on a classic autoencoder architecture. They apply W-Net, which is two assembled U-Net auto encoder networks, for multitask learning, where first block is assigned to semantic segmentation and the second one designed for unsupervised task.

In (Rao, De Deuge, Nourani–Vatani et al. 2017), authors propose two multimodal learning algorithms for obtaining commonality between visual imagery and acoustic bathymetry datasets. The first architecture used for classification task involves separate feature learning layer for each modality in midlayer with Denousing Autoencoder for both and ScSPM technique for feature extraction for visual modality. Then it is followed by a common representation layer for learning high-level correlations between the two. The second architecture extends on the first, but where gated model is used for the common layer, which is argued to be better in capturing one-to-many relationships across two modalities. It can perform unsupervised clustering and predicting visual features for visually unobserved locations just based on acoustic bathymetry data.

Marine habitat mapping refer to creating a map which covers seabed containing habitats separated by clear boundaries, where habitat itself means physical and environmental conditions belonigng

to particular biological community. In (Brown et al. 2011), Brown, Craig J et al. review 3 main strategies of seafloor obtaining benthic habitat maps such as abitotic surrogate mapping, asemble first, predict later, and predict first, assemble later methods. The first strategy (abitoic surrogates) does not involve combining in situ biological and geological data with environmental data, it utilizes unsupervised classification apprach to find patterns of environmental data. The second strategy (assemble first, predict late) applies benthic habitat classification scheme to generate maps of generalized habitat classes, single species and community maps based on their geological and biological features. The third strategy (predict first, assemble later) is used to produce species distribution map by modelling the ground truth data of focal species as a function of environmental predictors.

In (Le 2013), author introduces a deep sparse autoencoder for face detection, which learns high-level class-specific features from unlabeled image dataset with or without human faces. The network's architecture consists nine layers, which is just three replications of local receptive fields, which connects features to small regions, local pooling, which makes it invariant to local deformations, and local contrast normalization.

In (Abdulazizov et al. n.d.), autoencoder model for feature extraction from combination of multi-beam echosound backscatter and bathymetry data is designed based on the works of Shields et al. 2020. The data from both sources was fed to the network as a two-channel image and the learnt encoding is thought to be fed to the habitat classifier, which was left as a future work by the authors.

In (Shields et al. 2021), the authors focuse on autonomous planning methods for comprehensive and representative AUV surveys to visit locations with unique habitats by effective exploration of feature space representation of the bathymetric data. They use Variational Autoencoder (VAE) with Evidence Lower Bound loss function to project the bathymetry data of the entire space into the bathymetry latent space, which are then used for the two planing methods of global optimisation representative points and continuous exploration planners.

In (Ahsan et al. 2012), the authors present the use of one of the parametric generative probabilistic models, Gaussian Mixture Models (GMMs) for benthic habitat mapping and compares its perforance with other popular methods such as Classification Trees and Support vector machines. The model learns the correlation between seabed bathymetry and habitat classes givent he limited amount of available sampled data. The results showed that GMMs perform better that classical approaches when the data is scarce, showing that they have low sensitivity to amount of training data and better certainty towards their predictions.

In (Yamada et al. 2022), authors introduce a novel semi-supervised learning method for georeferenced imagery that improves learning efficiency and facilitate annotating data for CNN based classifiers of natural scenes designed for environmental monitoring tasks. The overall model is composed of three parts: deriving latent space representations of unannotated imagery data by using location guided autoencoders(LGA) which is unsupervised learning part; assigning human annotated labels for a subset of representative imagery dataset based on hierarchical k-means clustering and algorithm-generated pseudo-labels for the rest which is prioritised labelling part; input the results from prioritised labelling into CNNs for training part for classification task. The introduced LGA driven semi-supervised model enables it to be efficiently applied on a per dataset basis in the areas where trasnferability of learning between datasets is limited. The method was evaluted in four various enviromental datasets, which included seafloor and aerial images, and exhibited accuracy improvements by a factor up to 1.5 with small number of annotations available.

# 4 Intended methods

## 4.1 Datasets

The initial objective of this study involves constructing and deploying the model on a dataset representing the Earth's terrestrial surface, comprising a Digital Elevation Model (DEM) and corresponding labels derived from satellite visual imagery. This choice is motivated by the availability of larger-scale data with ground truth annotations. Subsequently, successful model testing will pave the way for its adaptation to an underwater dataset encompassing bathymetry data and visual images.
In terms of data availability, the dataset is characterized by a significant proportion of bathymetry data, followed by visual images, which are several orders of magnitude smaller, and labels, which constitute the smallest fraction. The limited quantity of visual images and even scarcer availability of labels underscore the need to employ a self-supervised learning approach, which is explored in this study. The relative distribution of the available data components is depicted in Figure 3.

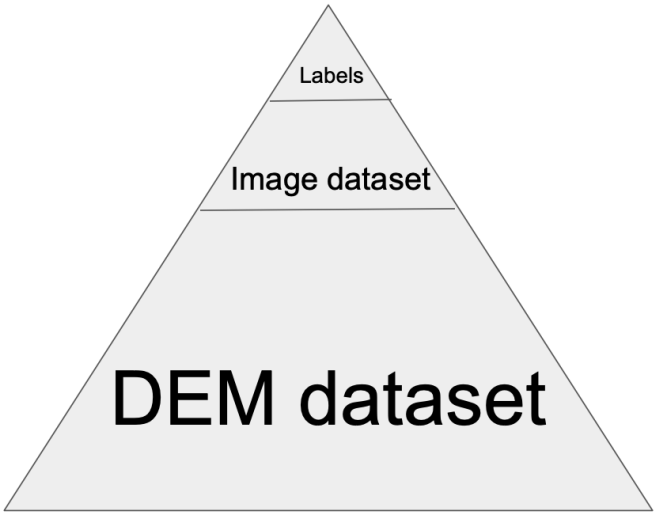



Figure 3: Possible distribution of data in the dataset.

### 4.1.1 Terrestrial dataset

This study employed two distinct datasets serving different purposes. The first dataset was specifically chosen for terrestrial applications. In selecting this dataset, certain criteria were considered essential. Firstly, the dataset needed to encompass an elevation model that exhibited sufficient variability to establish meaningful correspondences with bathymetry data and enable effective feature learning. Additionally, it was crucial for the dataset to provide corresponding labeled mappings for the designated area. Based on these requirements, Sweden emerged as the preferred candidate due to its topographical characteristics aligning with the specified criteria.

**Digital Elevation Model.** For the Digital Elevation Model (DEM) ALOS Global Digital Surface Model from Jaxa was used. The Japan Aerospace Exploration Agency (JAXA) created the JAXA Digital Elevation Model (DEM), a dataset that offers incredibly accurate and in-depth topographical data for the Earth's surface. The Advanced Land Observing Satellite (ALOS), which was launched by JAXA in 2006, provided the data used to create this dataset. The L-band Synthetic Aperture Radar (PALSAR) sensor, which was carried by the ALOS satellite, was used to gather radar data in order to produce the JAXA DEM. One of the finest resolution DEM datasets for worldwide coverage, the JAXA DEM has a spatial resolution of 30 meters. The dataset has a

vertical precision of roughly 5 meters and covers the whole planet, with the exception of specific places like the poles and small islands. It is an important resource for a variety of uses, such as natural resource management, environmental modeling, and geographic information systems. Data collection, calibration, and processing were all steps in the lengthy process that went into creating the JAXA DEM. Radar signals from the PALSAR sensor on the ALOS satellite were able to collect data on the Earth's surface because they passed through clouds and vegetation. The radar backscatter images produced by processing the PALSAR sensor's data were then used to produce interferograms. Interferograms are representations of the Earth's surface that depict variations in elevation over time by fusing two or more radar images acquired at separate times. The interferograms were produced, and then they underwent additional processing to produce a digital elevation model. In order to account for elements like air conditions, the curvature of the Earth, and data inaccuracies, the method included making a variety of corrections and changes to the data. The correctness of the generated digital elevation model was subsequently verified using ground control points and additional sources of elevation data. It is a useful tool for academics, scientists, and other professionals who need precise and thorough information about the topography of the Earth. It is especially helpful for applications like terrain analysis, flood modeling, and natural resource management due to its high spatial resolution and vertical precision (*ALOS Global Digital Surface Model "ALOS World 3D - 30m (AW3D30)"* 2023). The digital elevation model for the territory of Sweden consists of separate tiles of size 1800 by 3600 pixels with a resolution of 1 arcsecond (approximately 30 meters), which are latitude dependent. The examples of such tile can be seen in figure 4.

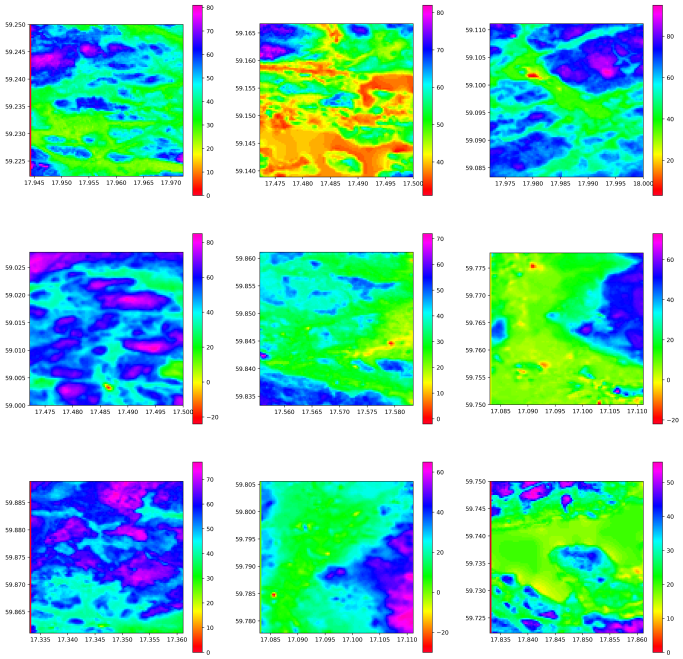


Figure 4: Examples of DEM tiles of Sweden. The horizontal x-axis coordinates are expressed in degrees longitude (Easting), while the vertical y-axis coordinates are presented in degrees latitude (Northing). The colorbar featured in the figures corresponds to the elevation height measured in meters relative to sea level.

**Labeled map of Sweden.** For obtaining labelled maps of the Swedish terriotory, Swedish National Land Cover Database was used. The Swedish University of Agricultural Sciences (SLU), the Swedish National Space Agency (SNSA), and other partners worked together to create the publicly accessible information known as the Swedish National Land Cover Database (NMD). For the purposes of land management, planning, and environmental studies, it offers a thorough and in-depth analysis of the various forms of land cover and how they have changed in Sweden. The entire Swedish area, including all land and water, is included in the NMD. It is made up of various different layers of data, such as land use, vegetation kinds, water bodies, and metropolitan areas.

These layers are based on high-resolution satellite imagery and additional data from sources including field surveys and aerial photography. The most recent update to the database, which is done every five years, covers the years 2017 to 2019. The NMD's high level of precision and detail is one of its distinguishing characteristics. The database's land cover classification system consists of 30 classes, which are further broken down into sub-classes for a total of 127 classifications. Also, there is a base map with 25 thematic categories which is divided into three hierarchical levels. This map is displayed as a raster with a minimum mapping unit of 0.01 hectare and a resolution of 10 meters. In addition to the main map, there are a number of additional layers that provide details on size and dimensions of objects, productivity, usage of land and forests in mountainous areas.The classification table can be seen in figures 5, 6 and 7.

| Value | Class | Definition |
|---|---|---|
| 111 | Pine forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10% where >70% of the crown cover consists of pine. Trees are higher than 5 meters |
| 112 | Spruce forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10% where >70% of the crown cover consists of spruce. Trees are higher than 5 meters |
| 113 | Mixed coniferous not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10% where >70% of consists of pine or spruce, but none of these species are >70%. Trees are higher than 5 meters. |
| 114 | Mixed forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10% where neither coniferous nor deciduous crown cover reaches >70%. Trees are higher than 5 meters. |
| 115 | Deciduous forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10% where >70% of the crown cover consists of deciduous trees (primarily birch, alder and/or aspen). Trees are higher than 5 meters. |
| 116 | Deciduous hardwood forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10 where >70% of the crown cover consists of deciduous trees, of which >50% is broad-leaved deciduous forest (mainly oak, beech, ash, elm, linden, maple, cherry and hornbeam). Trees are higher than 5 meters. |
| 117 | Deciduous forest with deciduous hardwood forest not on wetland | Tree-covered areas outside of wetlands with a total crown cover of >10 where >70% of the crown cover consists of deciduous trees, of which 20 - 50% is broad-leaved deciduous forest (mainly oak, beech, ash, elm, linden, maple, cherry and hornbeam). Trees are higher than 5 meters. |
| 118 | Temporarily non-forest not on wetland | Open and re-growing clear-felled, storm-felled or burnt areas outside of wetlands. Trees are less than 5 meters. |
| 121 | Pine forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10% where >70% of the crown cover consists of pine. Trees are higher than 5 meters |
| 122 | Spruce forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10% where >70% of the crown cover consists of spruce. Trees are higher than 5 meters |

Figure 5: Table with thematic categories 1 of a base map of NMD

Source: (*National Land Cover Database* 2023)

| Value | Class | Definition |
|---|---|---|
| 123 | Mixed coniferous on wetland | Tree-covered areas on wetlands with a total crown cover of >10% where >70% of consists of pine or spruce, but none of these species are >70%. Trees are higher than 5 meters. |
| 124 | Mixed forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10% where neither coniferous nor deciduous crown cover reaches >70%. Trees are higher than 5 meters. |
| 125 | Deciduous forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10% where >70% of the crown cover consists of deciduous trees (primarily birch, alder and/or aspen). Trees are higher than 5 meters. |
| 126 | Deciduous hardwood forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10 where >70% of the crown cover consists of deciduous trees, of which >50% is broad-leaved deciduous forest (mainly oak, beech, ash, elm, linden, maple, cherry and hornbeam). Trees are higher than 5 meters. |
| 127 | Deciduous forest with deciduous hardwood forest on wetland | Tree-covered areas on wetlands with a total crown cover of >10 where >70% of the crown cover consists of deciduous trees, of which 20 - 50% is broad-leaved deciduous forest (mainly oak, beech, ash, elm, linden, maple, cherry and hornbeam). Trees are higher than 5 meters. |
| 128 | Temporarily non-forest on wetland | Open and re-growing clear-felled, storm-felled or burnt areas on wetlands. Trees are less than 5 meters. |
| 2 | Open wetland | Open land where the water for a large part of the year is close by, in or just above the ground surface. |
| 3 | Arable land | Agricultural land used for plant cultivation or kept in such a condition that it can be used for plant cultivation. The land should be able to be used without any special preparatory action other than the use of conventional farming methods and agricultural machinery. The soil can be used for plant cultivation every year. Exceptions can be made for an individual year if special circumstances exist. |
| 41 | Non-vegetated other open land | Other open land that is not wetland, arable land or exploited vegetation-free surfaces and has less than 10% vegetation coverage during the current vegetation period. The ground can be covered by moss and lichen. |
| 42 | Vegetated other open land | Other open land that is not wetland, arable land or exploited vegetation-free surfaces and has more than 10% vegetation coverage during the current vegetation period. |

Figure 6: Table with thematic categories 2 of a base map of NMD

Source: (*National Land Cover Database* 2023)

| Value | Class | Definition |
|---|---|---|
| 51 | Artificial surfaces, building | A durable construction consisting of roofs or roofs and walls and which is permanently placed on the ground or partly or wholly below ground or is permanently placed in a certain place in water and is intended to be designed so that people can stay in it. |
| 52 | Artificial surfaces, not building or road/railway | Artificial open and vegetation-free surfaces that are not building or road/railway. |
| 53 | Artificial surfaces, road/railway | Road or railway. |
| 61 | Inland water | Lakes or water-courses. |
| 62 | Marine water | Sea, ocean, estuaries or coastal lagoons. |
| 0 | Outside mapping area | Outside the borders of Sweden and the Exclusive Economic (EEZ) Zone |

Figure 7: Table with thematic categories 3 of a base map of NMD

Source: (*National Land Cover Database* 2023)

This degree of specificity enables a more accurate and nuanced representation of the landscape and its historical changes. In order to evaluate the accuracy of the data, the NMD also provides details on the quality and confidence level of each categorization. The Swedish National Space Data Infrastructure (SNDI) portal, which offers access to a variety of geographic datasets, makes the NMD publicly downloadable. The database is made available in a variety of formats, including vector and raster data that is GIS-ready and web map services. The information can be applied to many different things, including detecting changes in land cover, studying climate change, and assessing biodiversity and landscape ecology. Academics, decision-makers, and practitioners engaged in land management, planning, and environmental studies can benefit greatly from the Swedish National Land Cover Database (NMD). It is a crucial tool for comprehending the terrain and how it has changed over time due to its high degree of detail, accuracy, and accessibility (*National Land Cover Database* 2023). The total map is comprised of tiles of different sizes examples of which are demonstrated in figure 8. All of the labeled tiles were combined to represent the whole labelled map of Sweden (Figure 9).

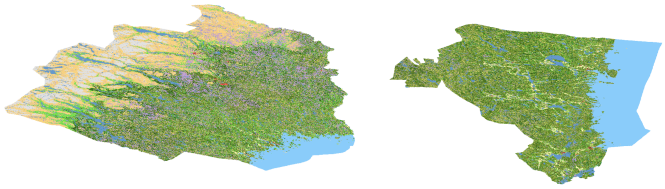


Figure 8: Labeled tiles of Sweden

Figure 9: Merged labeled tiles of Sweden

**Co-located data for terrestrial dataset.** To obtain matched data for terrestrial dataset, DEM patches of 100x100 pixels are obtained by dividing DEM tiles from Jaxa dataset. Two datasets have different coordinate reference systems (CRS). For example, DEM pathces are in EPSG:4326. It is a coordinate reference system (CRS) that utilizes latitude and longitude values to represent locations on Earth's surface. In this CRS, the equator serves as the point of reference for latitude, while the prime meridian, which runs through Greenwich, London, serves as the reference for longitude. For NMD, it's CRS is in EPSG:3006, or SWEREF 99 TM. It is a coordinate reference system employed in Sweden to precisely depict locations within the country. It serves as a spatial

reference framework, ensuring accurate representation of geographical points. The system utilizes meters as its unit of measurement, offering convenience for a range of applications such as surveying, mapping, and geospatial analysis. Large labelled imaged patches of Swedish territory were reprojected into corresponding coordinate reference system of DEM patches. After that, smaller labelled image patches were obtained for each DEM patch matching the same latitude and longitude coordinates and dimensions. Since their dimensions varied from DEM patches, they were rescaled from around 350x175 to 100 by 100 pixeled images. Overall, two datasets of different sizes were created, smaller one making up around 2300 DEM patches and corresponding segmentation images, and bigger one comprising of over 100 thousand DEM patches and corresponding segmentation images. Furthermore, the categories provided by NMD were grouped together into 9 different classes such as forests on wetland, forests not on wetland, open wetland, ararble land, water surfaces, artificial surfaces, other open land, non-forest on wetland, and non-forest not on wetland. The process of grouping these categories was guided by the hierarchical structure outlined in the NMD, as well as informed judgments based on similarity and proximity, as indicated in the table containing thematic categories. Figure 10 shows pairs of DEM and labeled images from the dataset.

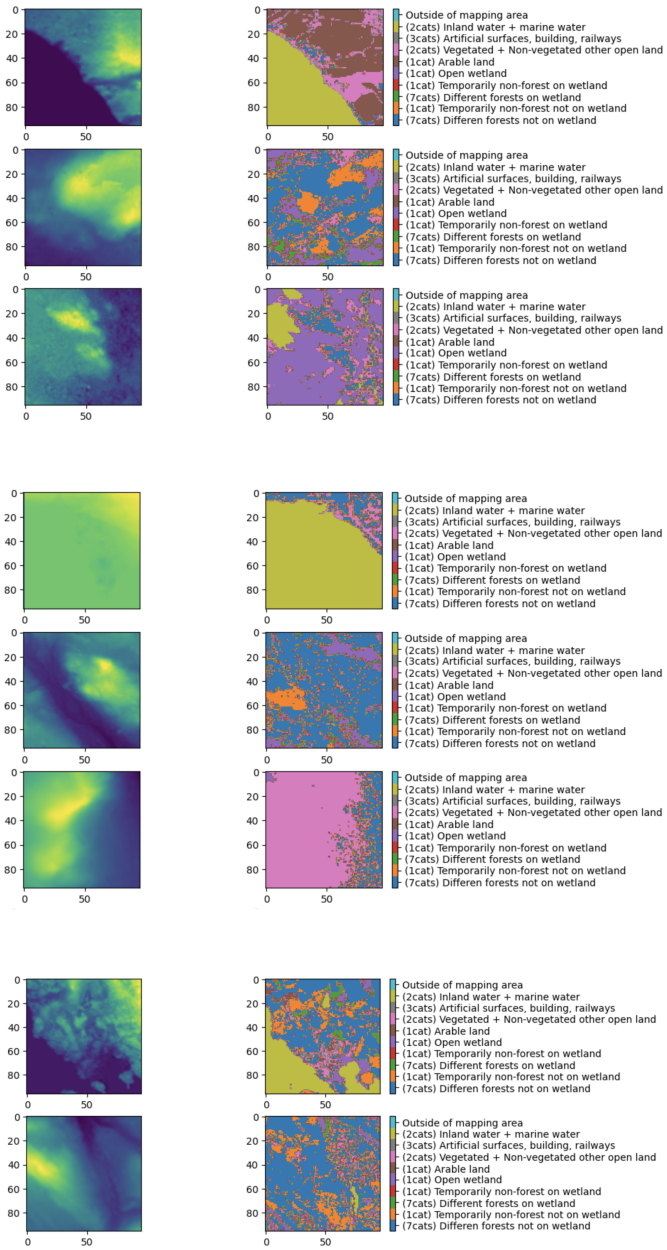



Figure 10: DEM and labeled image pairs from Swedish dataset

These two datasets extracted from ALOS Global Digital Surface Model and Swedish National Land Cover Database, respectively, was used for training the segmentation model and contrastive SSL model to establish standard metrics of how the system would perform on vast data with available corresponding labels.

### 4.1.2 Underwater dataset

The next step would involve evaluating the SSL model on the underwater data. The Southeastern Tasmania dataset, which was gathered in 2008 as a result of a partnership between The University of Sydney, The University of Tasmania, and Geoscience Australia (Spinoccia 2011; Williams et al. 2010), was utilized for the underwater dataset. The dataset and its acquisition process are described in the sections that follow.

**Bathymetry.** The scientific study of the topography and depth of oceans and other vast bodies of water is known as bathymetry. Typically, a ship's onboard sonar equipment, such as multi-beam echosounder (MBES), is used to collect bathymetric data. A transmitter and receiver are fastened to a ship's underbelly in order for an MBES to function. The sonar receiver picks up the 'pings' — a series of sound pulses—that the sonar head initially sent out and received after they were reflected off the ocean floor. The distance to each place on the ocean floor can be calculated by timing how long it takes for each pulse to travel there and return (*How Multibeam Sonar Works* 2009). This procedure is repeated repeatedly as the ship goes forward to map out a section of the bottom that corresponds to the sonar's sweep width. Detailed maps of the ocean floor can be made using the resulting bathymetric data, which is helpful for a variety of applications, such as ocean navigation, geological research, and oceanographic investigations. Bathymetric data is often processed after it has been collected by removing anomalies and merging measurements from the same place. A 2.5D Digital Elevation Map (DEM), which captures the depth of the ocean at each place in a two-dimensional matrix, is finally created from the bathymetric data. Figure 11 exhibits bathymetry map of the underwater seafloor at the Southeastern Tasmania. This bathymetric data of the interest is Geoscience Australia's vast gridded data. The 5 to 104 meter depth range is covered by a regularly spaced grid with 1.6 meter grid point intervals. The grid was produced by using a Simrad EM3002(D) 300kHz MBES system to analyse bathymetric data gathered by the research vessel Challenger in 2008 (Spinoccia 2011).

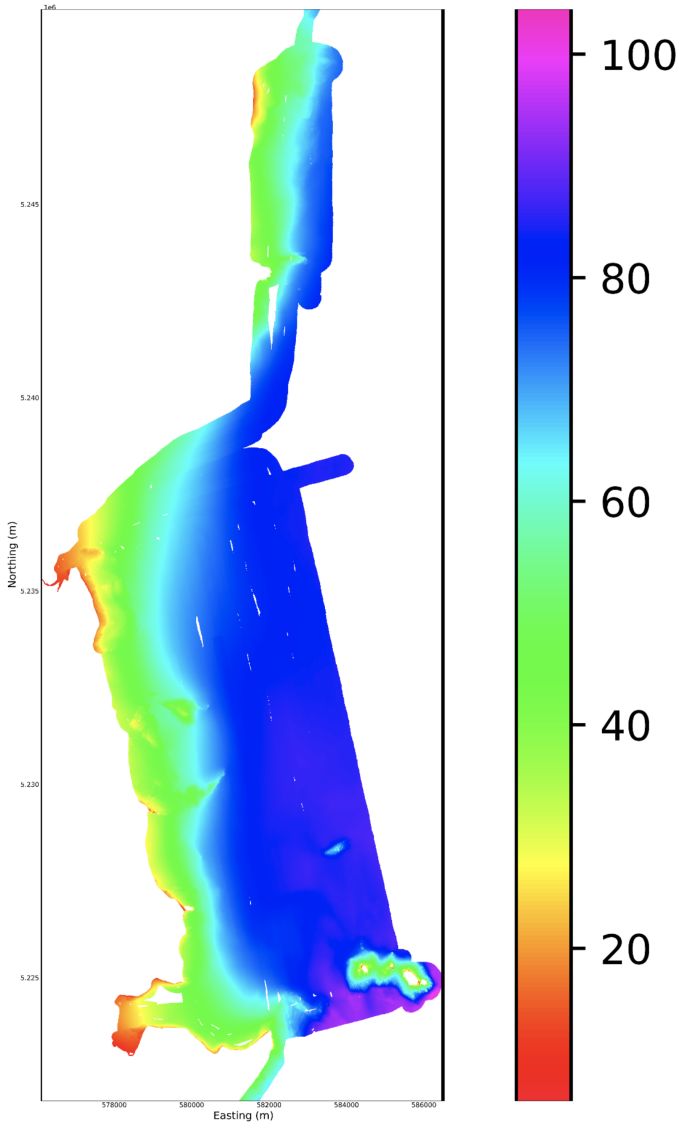



Figure 11: The bathymetry map over the entire Southeastern Tasmania region. The colorbar on the rigth shows the depth range from 5m depicted in red to 104m depicted in pink

**Underwater Visual images.** The AUV Sirius's downward pointed stereo cameras took high-resolution pictures of the ocean floor for the visual dataset (Williams et al. 2010). The photographs, which were taken on 11 dives and have a resolution of 1360 by 1024 pixels, depict a variety of ecosystems, including kelp forests and flat sandy areas. The majority of the photographs were shot at a height of 2 meters over the seafloor; photographs taken at higher altitudes appear dark, whereas photographs taken at lower altitudes are extraordinarily bright. The image labels for the eight different habitat classes were provided by expert annotations. Due to labeling mistakes and actual uncertainty within the fine-grained habitat groups, many labels were noisy. Figures 12 and 13 show RGB images of underwater terrain. Figure 14 represent the AUV dive path superimposed on the bathymetry data map.

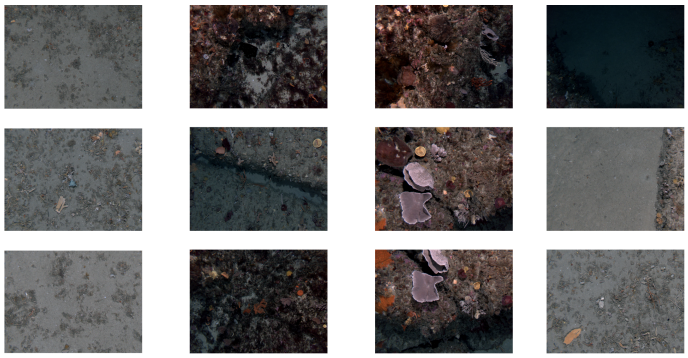


Figure 12: RGB image samples for the class labels between 1 and 4 presented per column. Between some habitat classes, there is visual ambiguity, and there is also some labeling noise.

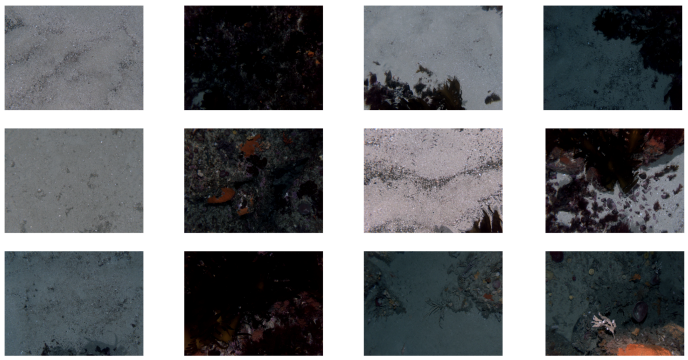


Figure 13: RGB image samples for the class labels between 5 and 8 presented per column. Between some habitat classes, there is visual ambiguity, and there is also some labeling noise.

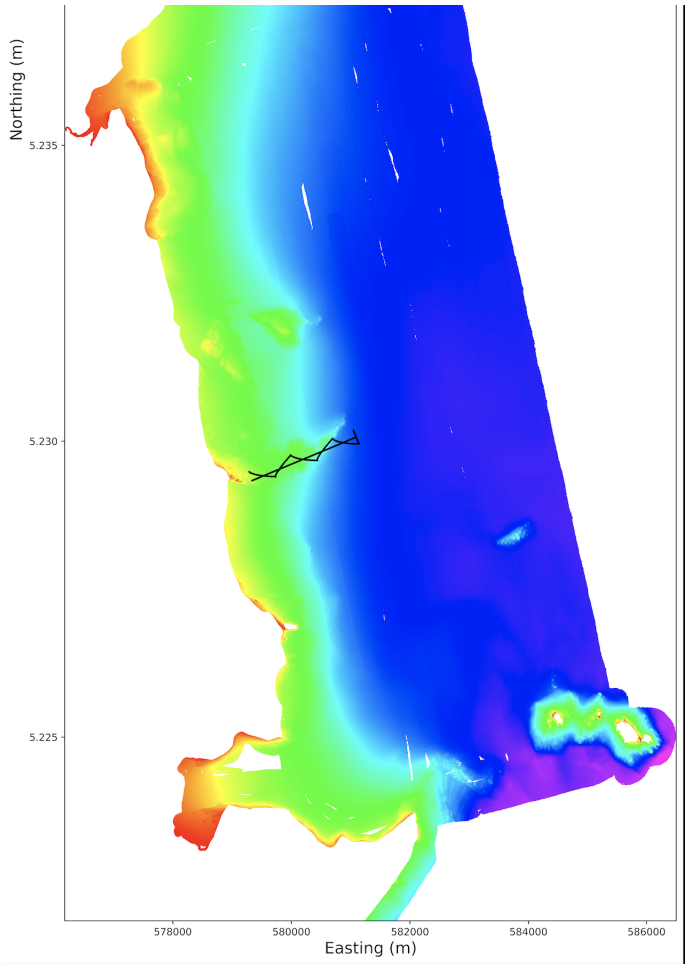



Figure 14: Survey path of the AUV obtaining underwater terrain photographs with corresponding labels. Path depicted in black on the bathymetry map.

**Co-located multimodal data for Underwater dataset.** As illustrated in 15, the multimodal data is created by extracting a 16 x 16 bathymetry patch and a 32 x 32 bathymetry patch that are both centered at the AUV point for each image. Because the AUV's position did not exactly match up with the centers of the grid cells, the bathymetric patch values were derived by using linear interpolation in the grid. Due to the certain distance between grid points, each patch corresponds to an area of approximately 30 x 30 and 60 x 60 meters, respectively. It is important to remember that this area is considerably greater than the standard 2-3 $m^2$ that an acquired image covers. Therefore, a bathymetric patch that matches the footprint of the optical image would not effectively detect much local structure.

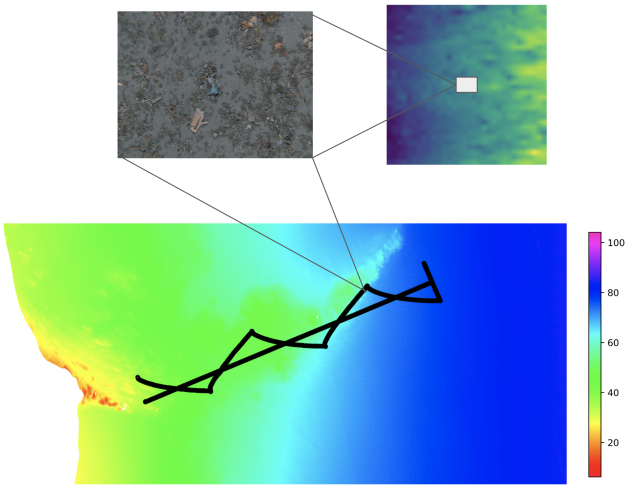



Figure 15: An example demonstrating the matching of photo (approx. bounding image radius under 1m) from an AUV survey to the appropriate bathymetry patch (30x30 meters) extracted from the whole bathymetry map. Pairs of 16 by 16 and 32 by 32 patches of gridded bathymetry are derived for each picture point along the AUV route which is depicted in black. The patch extends much beyond the size of the image. The colorbar on the right shown depth in meters for the bigger bathymetry map.

The size of the patch is determined by two criteria: it must be big enough to capture enough texture in the bathymetry and small enough to prevent encompassing different habitat classifications. The method described in (Bender et al. 2012) uses multi-scale features up to a 50m 50m region, while the author of (Rao, De Deuge, Nourani-Vatani et al. 2014) uses 15x15 bathymetry patches. Errors in the AUV's localization could be a problem with multimodal matching. Despite this, the habitats of interest often exhibit significant variations at larger scales, and the precision of AUV navigation is similar to the spacing of the bathymetric grid (Rao, De Deuge, Nourani-Vatani et al. 2014). Therefore, it is safe to infer that any potential alignment issues between the images and bathymetry caused by localization difficulties have minimal influence on the relationship between these two modalities (Rao, De Deuge, Nourani-Vatani et al. 2014). Some samples from the complete multimodal dataset is represented in figure 16 and the complete one consists of around 11 thousand visual pictures with corresponding label, each matched with a bathymetric patch. Additionally, the pure dataset of just bathymetry patches of similar patch dimensions were extracted from the whole bathymetry image, which resulted in approximately over 100 thousand bathymetry patches without labels for the purpose of self-suprevised learning. There are also substantial amount of unlabelled visual images taken during other surveys inside the same bathymetry area, which is shown in figure 17. For these images, corresponding bathymetry patches of two sizes were extracted as well, for the purpose of self-supervised learning.

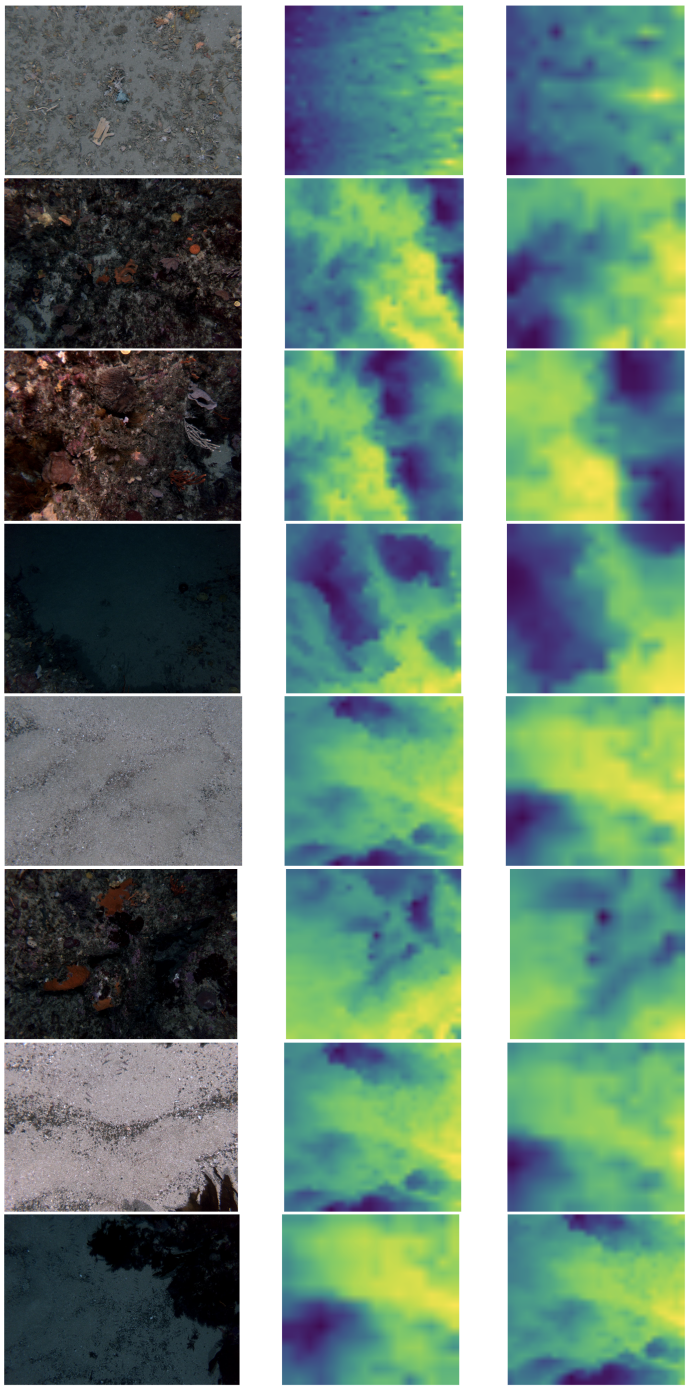


Figure 16: Instances of the marine data corresponding to various habitat classes between 1 and 8 are depicted in rows. Each image on the left part is paired with its corresponding bathymetric patches of sizes 16x16 and 32x32. The bathymetric patches cover a bigger area, measuring around 30 by 30 and 60 by 60 meters respectively, whereas the images often only catch an area of about 1.2m x 1.5m.

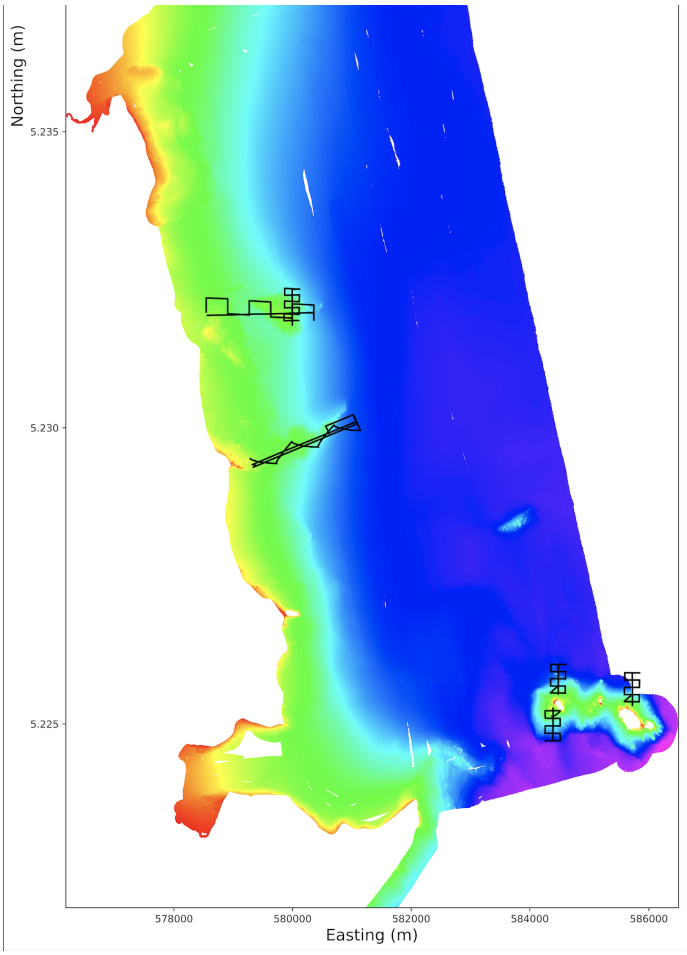



Figure 17: Different survey paths for visual image extractions depicted in black on the bathymetry map

## 4.2 Methods

To assess the effectiveness of contrastive learning on both terrestrial and underwater datasets, a systematic approach was adopted. Firstly, a terrestrial dataset which was manually constructed by consolidating diverse sources of digital elevation model and labelled map, ensured a representative collection of remote sensing data of Swedish territory. To establish a benchmark, a U-Net model was employed for segmentation on this dataset, providing a performance baseline for subsequent comparisons. Once the terrestrial dataset was adequately processed and analyzed, the contrastive learning method was applied, and the resulting performance improvements were evaluated. Further, the contrastive learning method was transferred to work on the underwater dataset which is the main focus of the study.

### 4.2.1 Segmentation - U-Net

The development of UNet, a novel convolutional neural network (CNN) architecture by Ronneberger et al (Ronneberger et al. 2015). in 2015, has significantly changed the area of picture segmentation. UNet has become a staple method in a variety of fields thanks to its extraordinary accuracy and precision. It has many uses, including in fields of the medical imaging (Ronneberger et al. 2015) in addition to autonomous driving (Giurgi et al. 2022) and satellite and aerial imagery(Rakhlin et al. 2018). The influence of UNet is noticeable in a variety of fields, where it continues to be crucial in obtaining insightful knowledge from intricate visual data. The encoder-decoder structure of the U-Net architecture's design enables it to successfully handle image segmentation tasks. The encoder component of the architecture downsamples the input image using convolutional and pooling layers, capturing crucial global features and contextual data. While adding skip connections, the decoder component reconstructs the segmented image using up-sampling techniques. In U-Net, the skip connections are essential. They create links between corresponding layers in the encoder and decoder, allowing information to be transferred and fine-grained details to be preserved. During the process of upsampling, this aids the model in maintaining precise segmentation borders and retaining crucial spatial information. The ability of U-Net to gather multi-scale contextual data is one of its main features. It increases segmentation accuracy and provides a better grasp of complicated images by mixing data from several levels of the network. Additionally, U-Net maximizes parameter usage, enabling quicker convergence during both training and inference. Due to its efficiency, it can be used in real-time applications where getting results quickly is crucial. The U-Net architecture for semantic segmentation includes both a contracting path and an expansive path. The contracting path follows the structure of a typical convolutional network. A rectified linear unit (ReLU) activation function is applied repeatedly to two 3x3 convolutions without padding, and then a 2x2 max pooling operation with a stride of 2 is used for downsampling. The number of feature channels doubles with each downsampling step. The expanding route consists of several steps, where each step starts with an upsampling of the feature map and ends with a "up-convolution" (also known as a 2x2 convolution) that cuts the number of feature channels in half. The appropriate cropped feature map from the contracting path is then concatenated with the upsampled feature map. Two 3x3 convolutions are then applied, with a ReLU activation coming after each one. In order to compensate for the border pixel loss that occurs during convolutions, cropping is required. Each feature vector, which has 64 components, is mapped to the desired number of classes using a 1x1 convolution in the architecture's top layer. There are 23 convolutional layers altogether in the U-Net architecture (Ronneberger et al. 2015). The overral achritecture of U-Net can be seen in figure 18

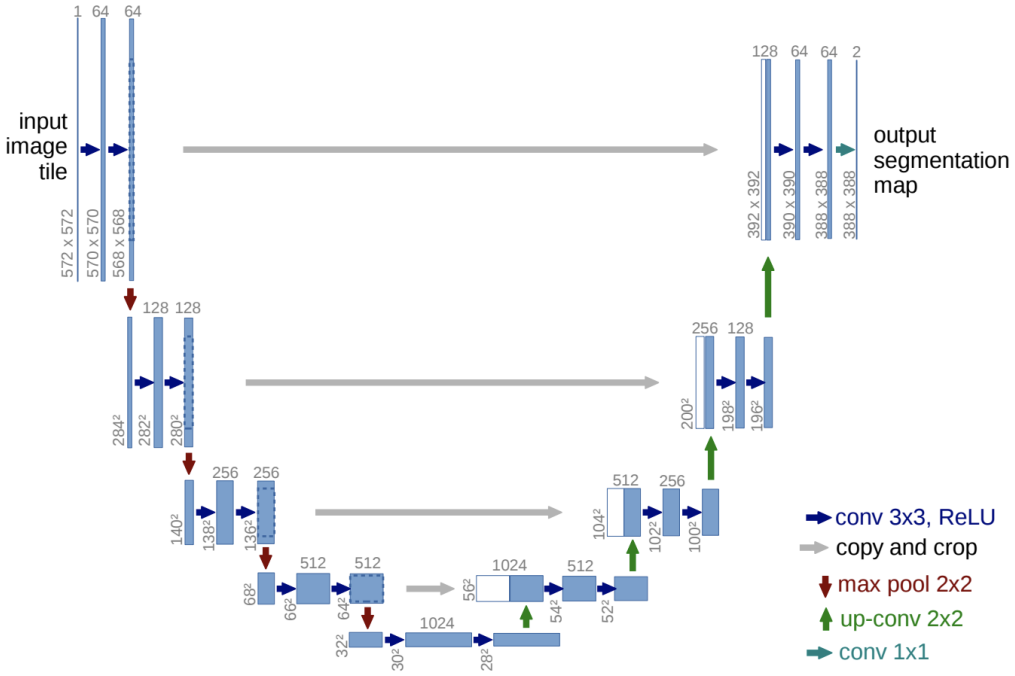



Figure 18: Overall U-Net architecture

Source: Ronneberger et al. 2015

### 4.2.2 Contrastive Feature Learning - SimCLR

Numerous cutting-edge approaches have recently surfaced in the field of self-supervised learning that are designed specifically toward image-based applications. When compared to supervised models, these developments have produced measurable performance improvements, especially when dealing with a shortage of labeled data. One of the novel methods include contrastive learning approaches. Contrastive learning techniques, more exactly, require training a model to efficiently group an image and its slightly transformed copy inside the latent space while also maximizing the dissimilarity to other images. SimCLR is a recent and simple method for reaching this goal. The general approach is to use provided unlabeled image dataset to train a model that can then quickly adapt to different image identification problems. A batch of image data is sampled in the usual way for each training iteration. Through the use of various data augmentation techniques including cropping, Gaussian noise, blurring, and others, two different copies of each unique image are created for each image. These augmentations are then processed using a convolutional neural network (CNN) architecture, such as ResNet, to produce a one-dimensional feature vector. These latent space representations are then taught to be similar, whilst all other image feauture representations in that batch should be as dissimilar as possible. In this manner, the model must learn to recognize visual content that is unaffected by data augmentations, such as objects that are typically of interest in supervised tasks. The implementation of SimCLR is made possible by using the data loader process with augmentations such as random cropping, grayscaling, gaussian blur, and color distortion. Two separate enhanced copies of an image, designated as xi and xj, are obtained at each iteration. The next step is to encode these photos into one-dimensional feature vectors with the aim of increasing their similarity to one another while decreasing their similarity to all other images in the batch. A base encoder network, designated as f(. ), and a projection head, denoted as g(. ), make up the encoder network. The deep convolutional neural network that serves as the base network is normally in charge of obtaining a representation vector from the enhanced data samples. The widely used ResNet-18 architecture is employed as f(.), and its output is referred to as f(xi) = hi. In order to compare similarity across vectors, the projection head g(.) transfers the representation vector h to a space where the contrastive loss is applied. A tiny Multi-Layer

Perceptron (MLP) with non-linearities is frequently used as the projection head. The projection head is described as a two-layer MLP in the original SimCLR paper, with ReLU activation in the hidden layer. The authors claim that wider or bigger MLPs can greatly improve performance in SimCLRv2. Since it was shown that increasing the MLP's depth resulted in overfitting on the provided dataset, we thus use an MLP with hidden dimensions that are four times larger. The rojection head g(.) will be dropped and f(.) will be used as a pretrained feature extractor after the contrastive learning training is complete. It has been seen that when fine-tuning the network for a new problem, the representations z produced by the projection head g(.) perform worse than those created by the base network f(.). This discrepancy might be attributable to the fact that the representations z are trained to become invariant to different properties, such as color, which may be essential for downstream tasks. Therefore, g(.) is only required at the contrastive learning stage. Regarding the training procudre, as previously stated, the goal is to maximize the similarity between two representations of the same image that have been enhanced, designated as zi and zj, while simultaneously limiting their similarity to all other examples in the batch. The illustration of SimCLR method is shown in figure 19.

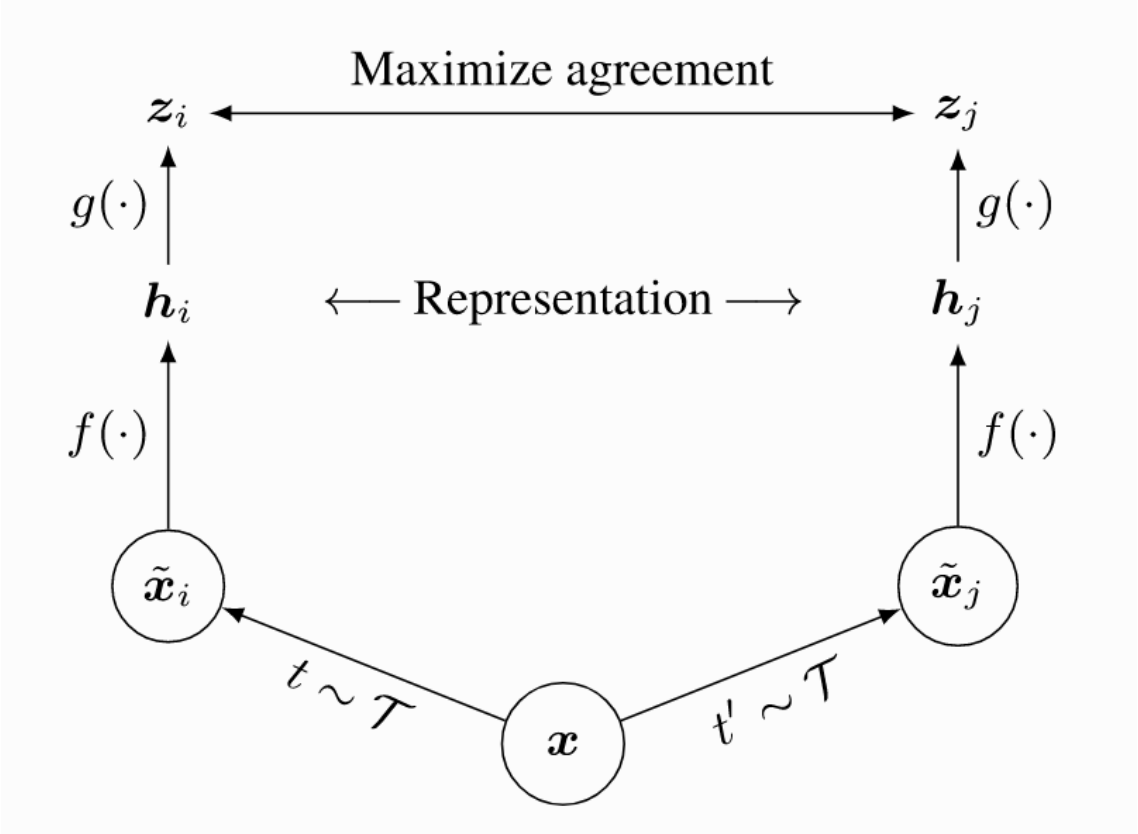



Figure 19: SimCLR network setup

Source: Chen et al. 2020

SimCLR uses the Information Noise-Contrastive Estimation (InfoNCE) loss, which was first developed by Aaron van den Oord et al. for contrastive learning, to do this. By applying a softmax function to the similarity data, the InfoNCE loss essentially compares the similarity of zi and zj to the similarity of zi with any other representation inside the batch. The loss is formally expressed as follows:

$$l_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{k \neq i} \exp(\text{sim}(z_i, z_k)/\tau)} = -\text{sim}(z_i, z_j)/\tau + \log \left[ \sum_{k=1}^{2N} \mathbb{1}_{k \neq i} \exp(\text{sim}(z_i, z_k)/\tau) \right]$$

The hyperparameter "tau" acts as the temperature parameter, controlling the concentration of the final distribution, while the function sim represents a similarity measure. The effect of several

different image patches can be changed in comparison to a single comparable patch by using a temperature parameter. The similarity metric selected for SimCLR is cosine similarity, which is defined as follows:

$$\text{sim}(z_i, z_j)) = \frac{z_i^{\text{T}} \cdot z_j}{\|z_i\| \cdot \|z_j\|}$$

The range of cosine similarity is between -1 and 1, with -1 denoting the least potential similarity and 1 the most. The characteristics of two different photographs typically converge towards a cosine similarity value that is near to zero. This tendency results from the requirement that zi and zj be exactly opposite in direction across all feature dimensions in order to achieve a cosine similarity of -1, which drastically limits the flexibility and variety of the representations (Chen et al. 2020).

## 4.3 Approach

This thesis work adopts a self-supervised multimodal learning approach to establish correlations between visual and underwater acoustic data, enabling them to complement each other. The proposed methodology involves the separate extraction of latent representations from bathymetry data and visual data through self-supervised learning techniques (approach 1). These pre-trained networks are then utilized to obtain their respective latent spaces, which are subsequently employed to train a classifier capable of categorizing different habitat classes. It is important to consider the tradeoff between the coverage provided by bathymetry and visual images. While the optical images offer higher discrimination capabilities, their coverage is limited to a small portion of the seafloor. Conversely, bathymetry, as a form of remote sensing, covers larger areas but exhibits lower resolution, potentially resulting in inferior feature representations. The overall model architecture is presented in 20.

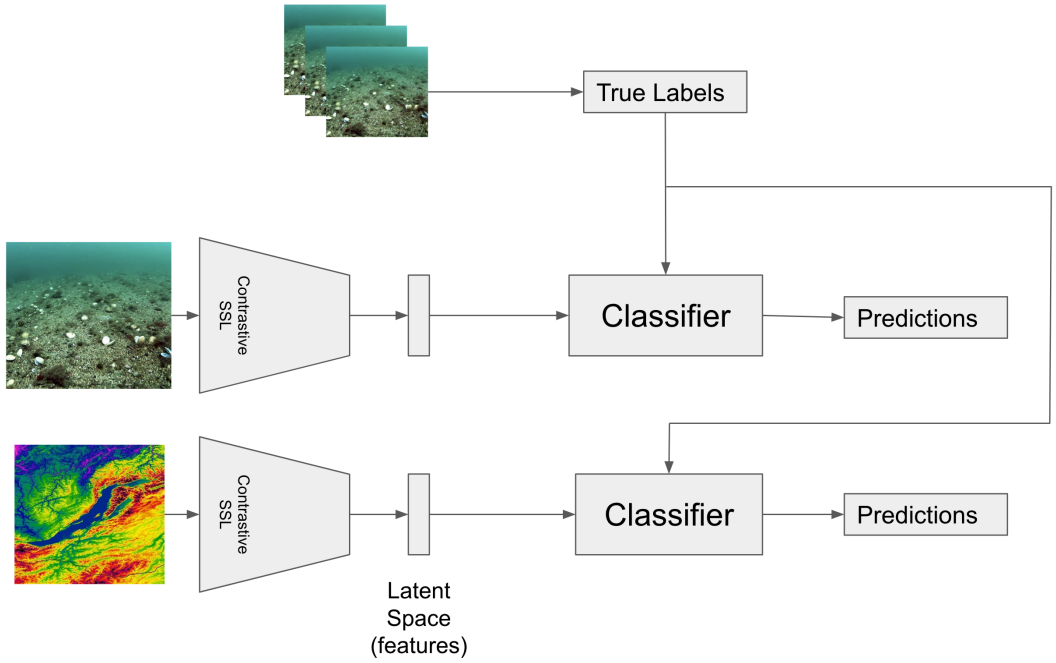



Figure 20: Overall model for separate training of bathymetry and optical images.

Other thing to compare, would be to simultaneously train bathymetry and optical images for self-supervised learning task (approach 2). By feeding to model pairs of acoustic image and optical image corresponding for the same positions, the model could learn to project high-quality features of optical images to acoustic ones, and hence, increase the performance of predicting class labels from bathymetry latent space representations. The overall model of paired training is presented in figure 21.

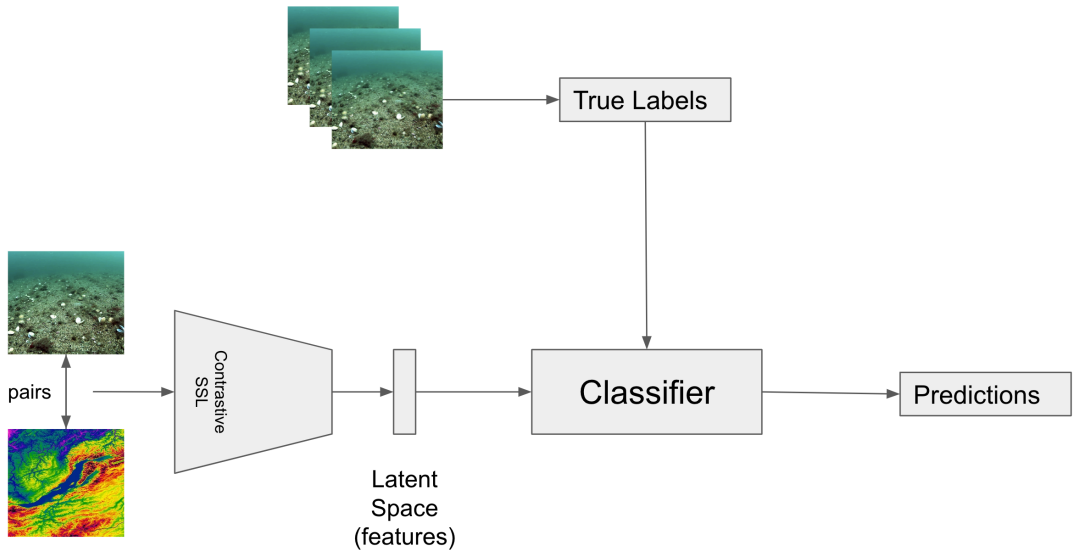



Figure 21: Overall model for synchronous training of bathymetry and optical images pairs. The presence of a double arrow connecting the bathymetry patch and visual image signifies their co-registration as paired data.

Next strategy, which is approach 3, would be to find out if the same approach of paired training would work if additionally to the paired data, separate bathymtry data would be also fed to the SSL model. The limited quantity of optical images might give the network rich details and make it correspond to bathymetry data of the same quantity, while additional supply of large unlabeled bathymetry data would make it able to generalize to the whole seafloor. The overall model of non-paried and paired training is presented in figure 22.

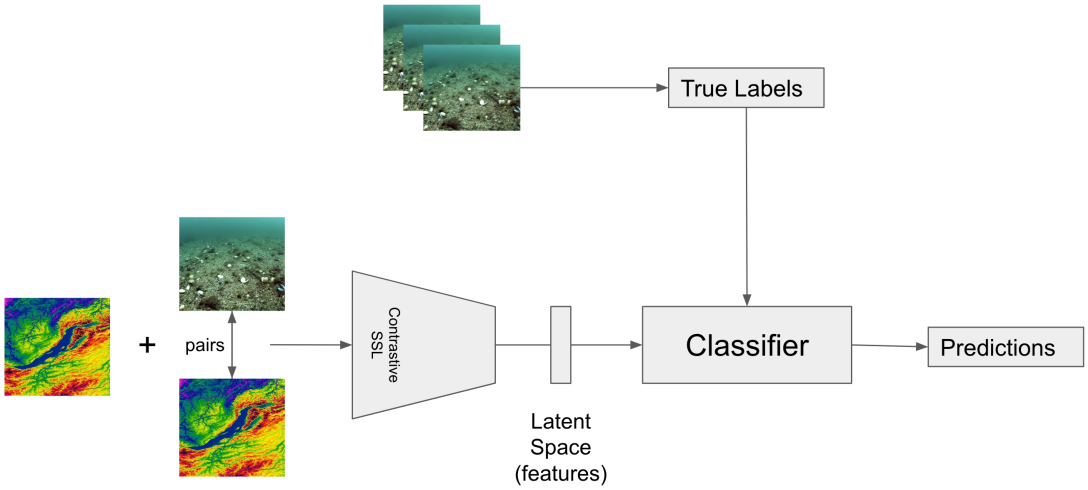



Figure 22: Overall model for synchronous training of single bathymetry plus bathymetry and optical images pairs. The presence of a double arrow connecting the bathymetry patch and visual image signifies their co-registration as paired data.

The final approach 4 will consist of simultaneous training of two networks connected via their loss. In this case, it would not be required to transform optical images into bathymetry patch size, and unlike using greyscale of acoustic images, three channel RGB information could be feed instead

into the separate SSL network, thus conserving more information. During the training, the losses from each of the nerworks would be added and back-propagagated thtrough the networks. The overall model of non-paried and paired training is presented in figure 23.

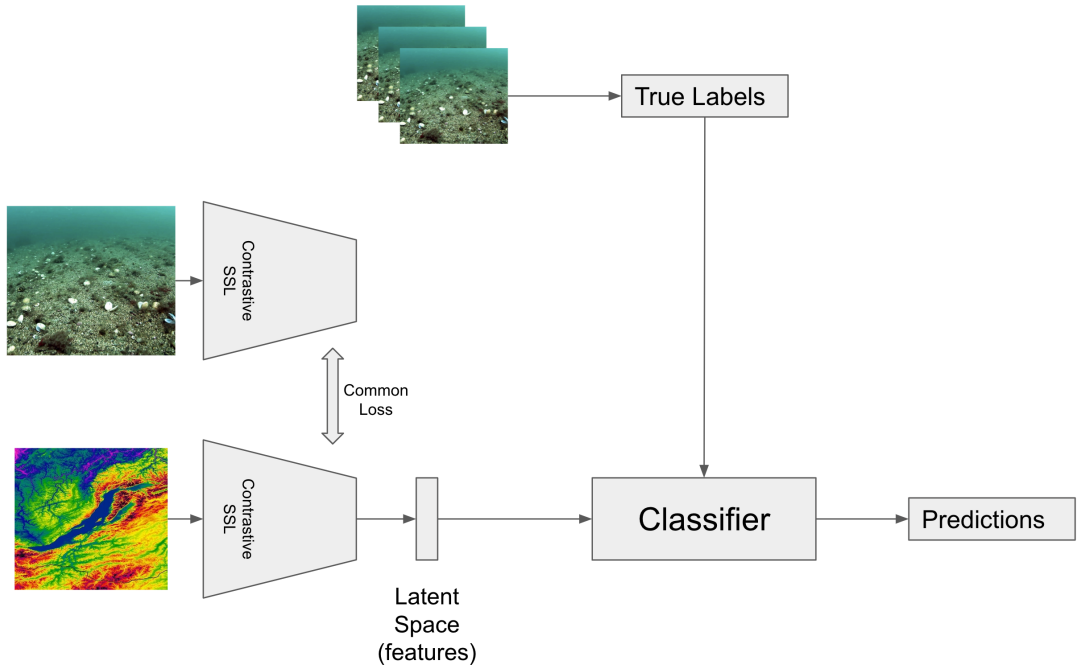



Figure 23: Overall model for synchronous training of single bathymetry plus bathymetry and optical images pairs with common loss.

# 5 Results

## 5.1 Results - Terrestrial dataset

### 5.1.1 Experiment and testing

The dataset employed for this purpose encompassed 2300 tiles along with their corresponding segmented images. The dataset was randomly split to train and test set with 85% and 15%, respectively. The class distributions for each are shown in figure 24. As can be seen from the figure, class 'Different forests not on wetland', which is made of 7 categories, constitute the majority of the distribution, followed by a 'Open wetland class'. Other classes make up comparatively smaller proportion of the dataset, while 2 other classes such as 'Temporarily non-forest on wetland' and 'Arable land' comprise tiniest insignificant part. It is clear from the dataset's class imbalance that some classes are vastly underrepresented in comparison to others. During the creation and assessment of the model, the impact of the class imbalance found in the dataset will be taken into consideration.

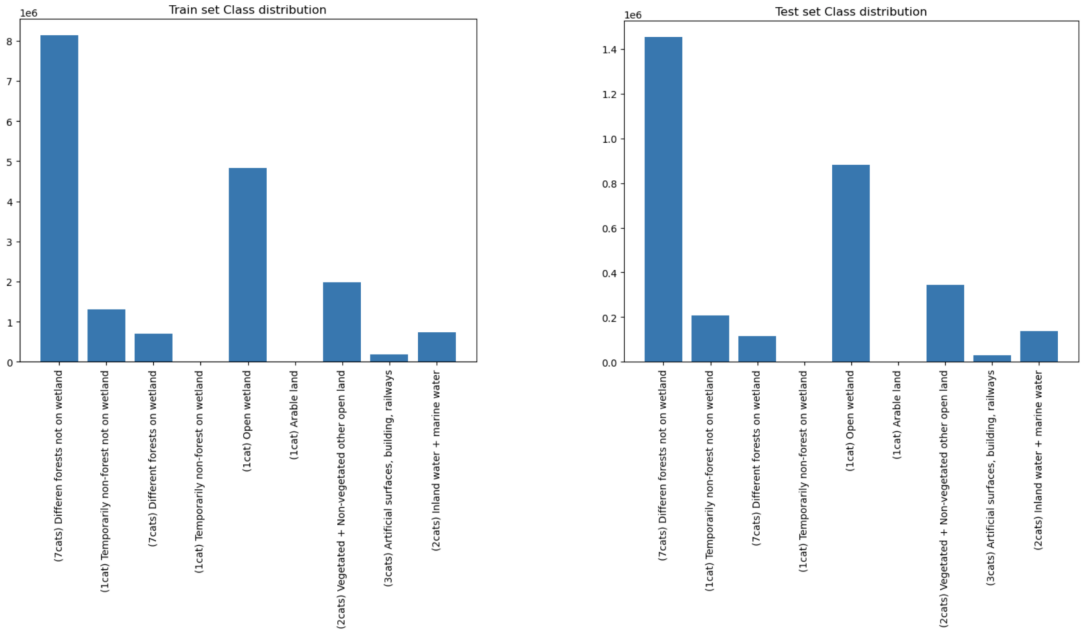



Figure 24: Swedish dataset class distribution

### 5.1.2 Segmentation

Initially, a UNet model was constructed and trained using terrestrial data derived from the Swedish territory. The training process spanned 1000 epochs, during which diverse hyperparameters, including learning rate and batch size, were systematically varied and evaluated. The training results are shown in figure 25.

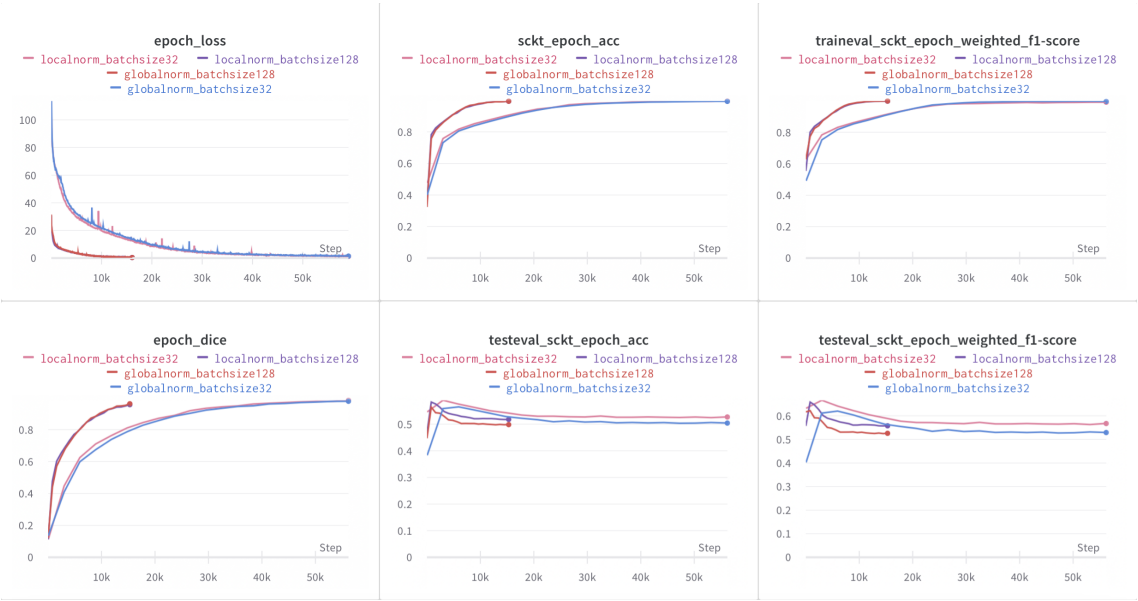



Figure 25: U-Net training results for Sweden Dataset

The training involved four versions of a segmentation model with local normalization along with batch size 32 and batch size 128, and global normalization along with batch size 32 and batch size 128. Local normalization refers to the normalizing the values in the image between 0 and 1 by subtracting lowest value from the image and dividing by the difference between highest and lowest values. the formula is given below:

$$normalized\_image = \frac{image - min\_image}{max\_image - min\_image}$$

Global normalization refers to subtracting from each image the highest value from the all available images and dividing it by the difference between highest and lowest values from the all available images. The formula is following:

$$normalized\_image = \frac{image - globalmin\_image}{globalmax\_image - globalmin\_image}$$

The findings depicted in the figure demonstrate a noteworthy training accuracy, achieving an exceptional value of 99% in terms of pixel accuracy. However, this impressive performance may potentially indicate the presence of overfitting, as the corresponding results on the test set yielded a comparatively lower pixel accuracy of approximately 60%.
To gain visual insights into the predictions generated by the trained model, Figure 26 presents visualizations of the model's predictions for a subset of samples from the training set. These visualizations reveal that the segmented images predominantly exhibit hues of red and orange, corresponding to the 'Different forests not on wetland' and 'Open wetland' classes. Additionally, the predictions closely align with the label structures observed in the ground truth images. Notably, there exist strong correlations between the patterns in the segmented images and the DEM patches for certain classes such as 'Open Wetland' and 'Different forests not on wetland', while weaker correlations are observed for other classes.

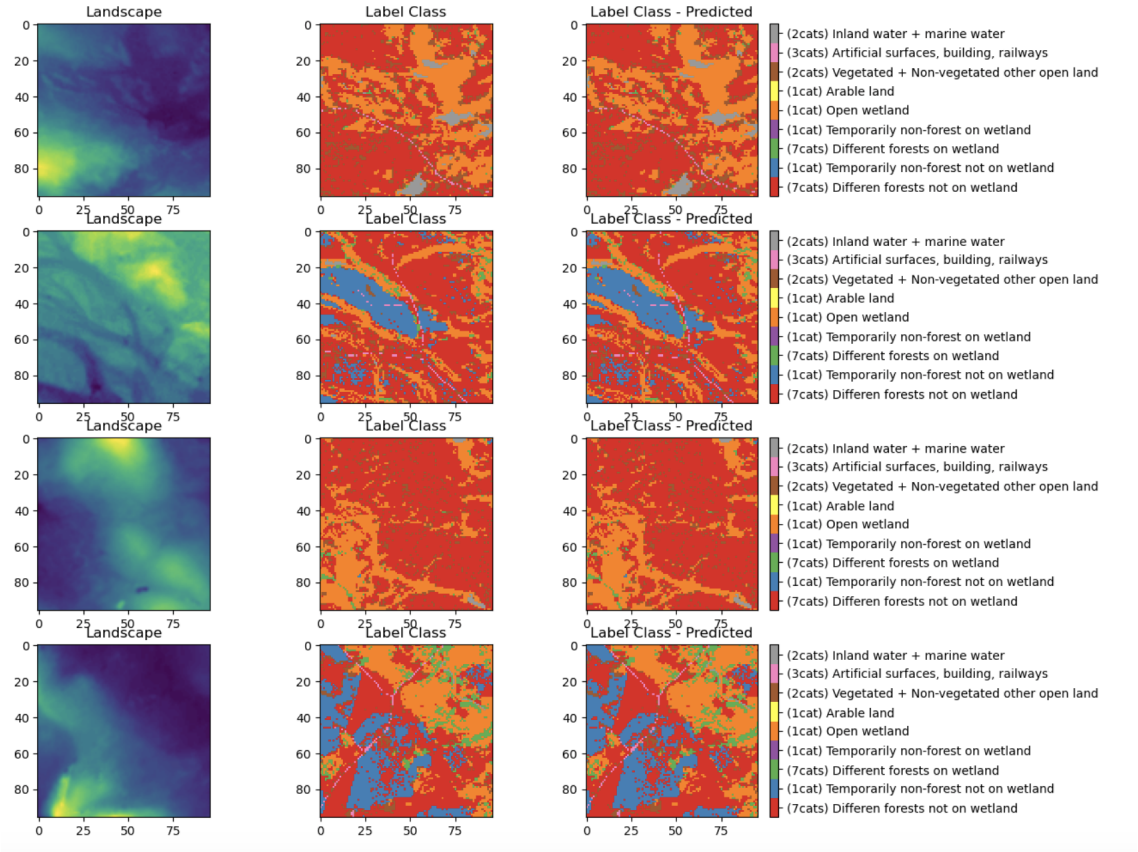



Figure 26: Visualization of U-Net predictions on train set

Likewise, Figure 27 portrays visualizations of the model's predictions for selected samples from the test set. The visualizations of the test set samples exhibit a similar trend, with the segmented images predominantly composed of the two prominent classes mentioned earlier. It is evident that, when not considering pixel-perfect accuracy, the predicted images generally align with the ground truth patterns derived from the DEM patches.

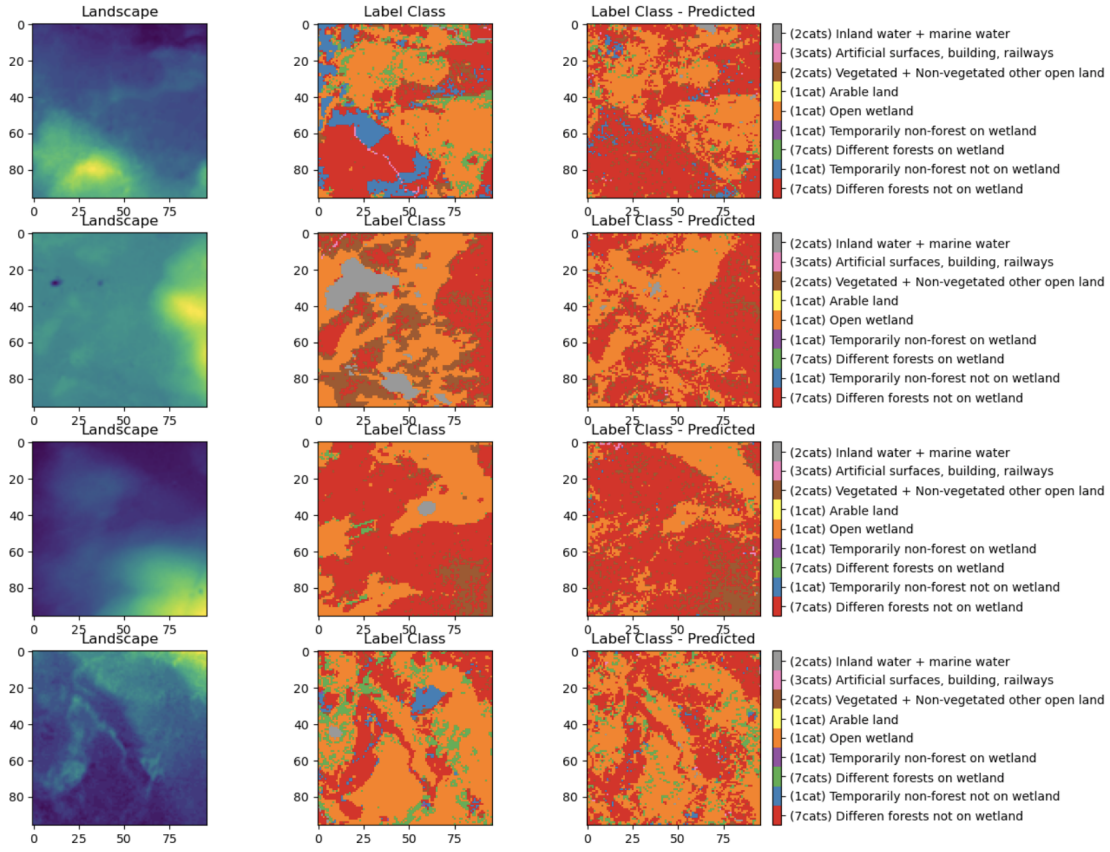



Figure 27: Visualization of U-Net predictions on test set

Additionally, the confusion matrices for the predictions on the training and test sets are presented in figures 28 and 29 respectively. The left side of the matrices represents the true labels, while the bottom part represents the predicted labels. It is evident that the confusion matrix from the training set exhibits a structure closely resembling an identity matrix, where the diagonal elements are 1 and the remaining elements are 0. This observation can be attributed to overfitting, thereby highlighting the model's excessive adaptation to the training data.
Contrarily, the confusion matrix obtained from the test set does not exhibit a similar identity matrix structure. Upon closer inspection, it becomes apparent that the first class, namely 'Different forests not on wetland' is predicted as the first class in 75% of cases, while 13% of instances are incorrectly predicted as 'Open wetland'. Furthermore, the 'Open wetland' class is correctly predicted in 63% of cases, while in 26% of instances, it is misclassified as 'Different forests not on wetland'. The remaining classes display a predominance of incorrect predictions, mainly oscillating between the 'Different forests not on wetland' and 'Open wetland' classes. This behavior can be attributed to class imbalance, as these two classes account for less than 80% of the overall class distribution, in addition to the overfitting exhibited on the training set.

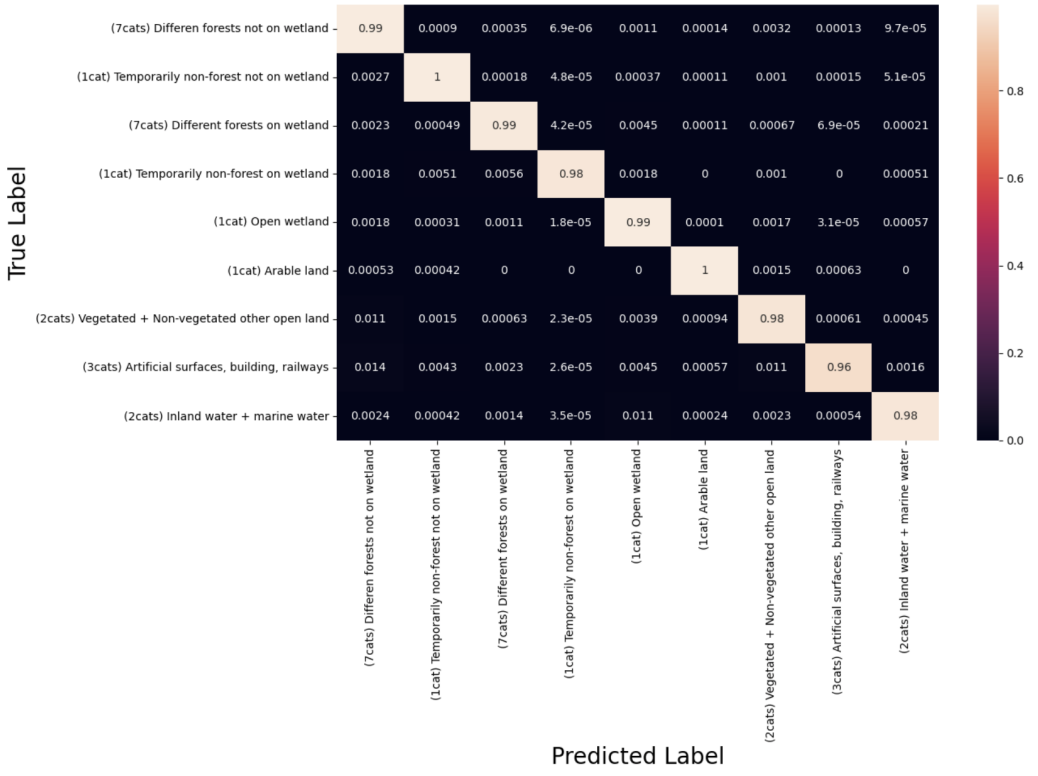



Figure 28: Confusion matrix from U-Net predictions on the train set

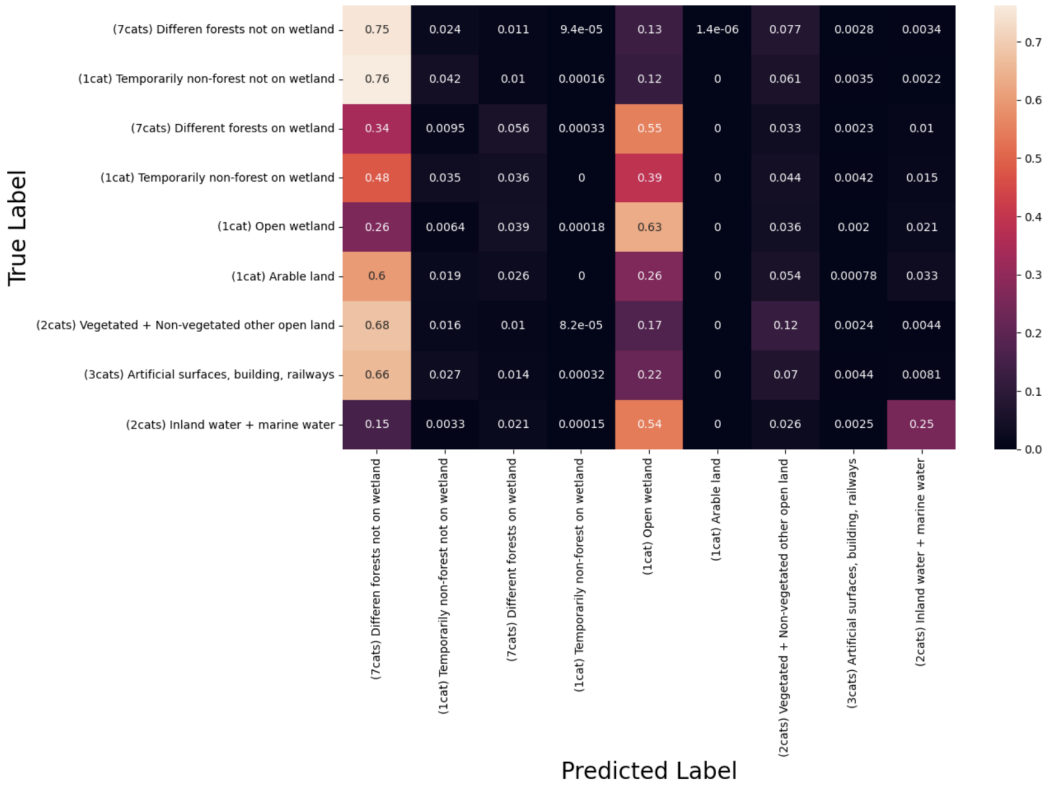



Figure 29: Confusion matrix from U-Net predictions on the test set

Having established the performance baseline using the U-Net segmentation model, the subsequent

objective is to assess the efficacy of contrastive learning on the identical dataset.

### 5.1.3 Contrastive learning

Following the evaluation of the segmentation model's performance on the custom dataset, the subsequent phase involved the implementation of a contrastive self-supervised learning model. The selected model for this purpose was SimCLR, which stands for Simple Contrastive Learning Framework. It was trained on DEM patches obtained from the Sweden dataset for a total of 500 epochs. The pretrained model was then frozen, and its feature extractor was employed to derive feature representations of the DEM patches. These representations were subsequently utilized in training a simple regression classifier. To provide labels for the training process, a single label per image was assigned by extracting the most frequent class label within each image patch. These labels were then incorporated into the model alongside the corresponding DEM patches. The results of the training process are presented in 30.


Figure 30: Classification results from downstream task

The evaluation of the downstream classification task reveals a noteworthy achievement, wherein the utilization of the contrastive learning model yields an approximate accuracy of 75%. This outcome underscores the efficacy of the contrastive learning model and underscores its potential for application in the subsequent analysis of the underwater dataset, augmented by the integration of multimodal training techniques.

## 5.2 Results - Underwater dataset

Following the successful testing and validation of the model on the terrestrial dataset, its application was extended to the underwater dataset. Analogous to the training process employed for the Swedish dataset, the SimCLR model underwent training on a substantial quantity of unlabeled bathymetry patches for 300 epochs, utilizing various hyperparameters such as learning rates and batch sizes. The pre-trained model successfully extracted the latent space representations of the bathymetry patches using the available corresponding labels after the training phase. These latent space representations were then utilized as inputs for training a regression classifier.

### 5.2.1 Experiment and testing

The distribution of classes among the 11,000 labeled images is presented in Figure 31. It is evident from the figure that class 2 labels account for a substantial proportion of the dataset, comprising approximately 37%. Following class 2, class 1 labels make up around 22% of the dataset. Classes 5, 6, 7, and 8 collectively constitute the remaining portion, albeit with smaller proportions, while class 4 and class 3 exhibit the smallest representation. This distribution shows that there is a significant class imbalance in the dataset, with some classes being significantly underrepresented relative to others. Consequently, the impact of this class imbalance will be considered during the development and evaluation of the model.

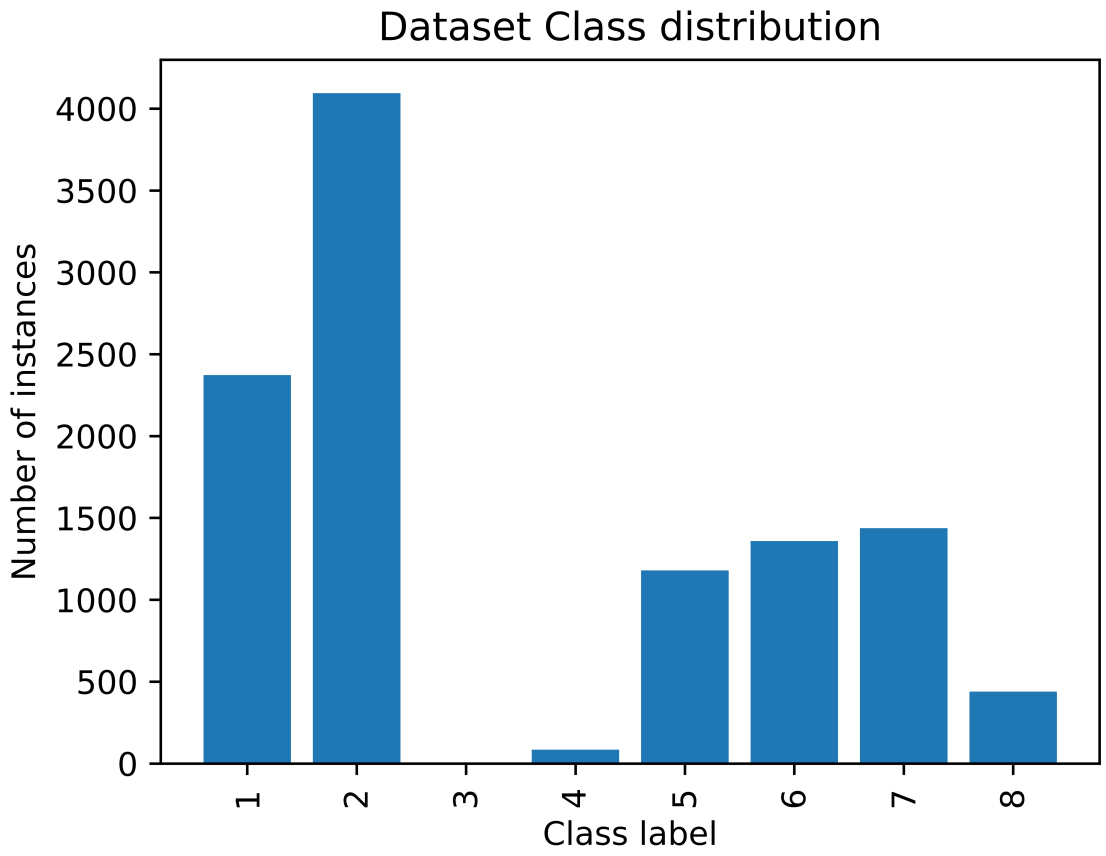



Figure 31: Underwater labelled dataset class distribution

Figure 32 illustrates the camera path overlaid with the corresponding class labels. The Autonomous Underwater Vehicle (AUV) path encompasses both straight lines and zigzag lines, with the zigzag lines intersecting the straight path. For model testing purposes, two approaches will be employed. The first approach involves randomly dividing the labeled dataset into training and testing groups of 90% and 10%, respectively. In contrast, the second approach, referred to as the 'line split' utilizes line labels, which account for 40% of the labeled dataset, for training, while the remaining 60% comprising zigzag labels are used for testing.

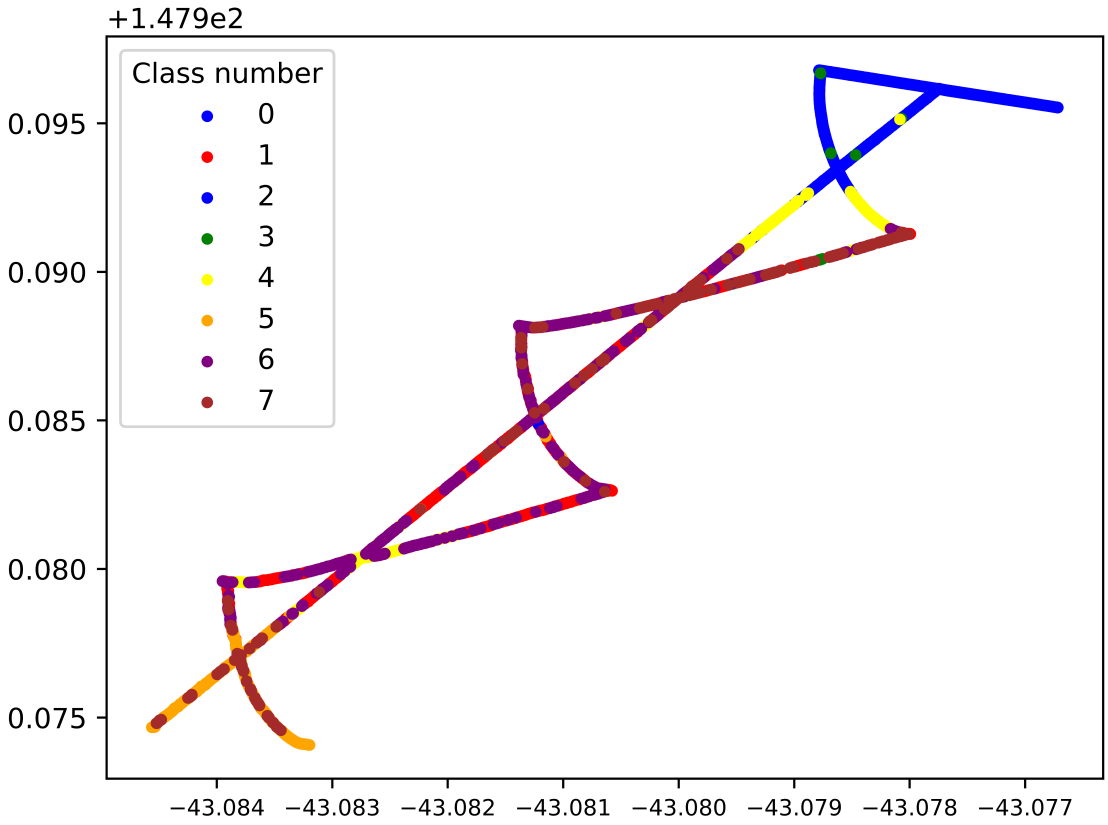



Figure 32: Class labels superimposed on the path of the AUV. The horizontal axis coordinates are presented in degrees Latitude, and the vertical axis coordinates are preseneted in degrees Longitude.

### 5.2.2 Bathymetric Feature Learning

#### Feature Learning

The approach 1 related to bathymetric feature learning outlined in Figure 20 was employed for this case. The SimCLR model was executed on bathymetric patches of sizes 16x16 and 32x32. A batch size of 256 was chosen for training, as larger batch sizes were observed to enhance the model's training process. The model was trained for 300 epochs, requiring approximately 4 days to complete.
The outcome of the training procedure is illustrated in Figure 33. The training progress exhibited a similar pattern for both patch sizes. The training loss steadily decreased over time, while the validation loss showed a fluctuating behavior with an initial increase followed by a subsequent decrease, demonstrating a considerable level of noise. The top-1 accuracy metric on the training set exceeded 80%, while on the validation set, it varied between 60% and 80%.
The top-5 accuracy metric is often preferred over top-1 accuracy as it provides a less noisy measure. It quantifies the frequency at which the correct image patch appears among the top-5 most similar samples within a batch. For the top-5 accuracy, the model achieved 98% on the training set and a similar level on the validation set.


Figure 33: SimCLR for latent representation learning from bathymetry patches of sizes 16x16 and 32x32

### Classification

The feature representation obtained from contrastive learning for bathymetry patches is subsequently utilized for classification purposes using Logistic regression. The training procedure involves training on patches of sizes 16x16 and 32x32, with corresponding batch sizes of 128 and 64, as illustrated in Figure 34.
The training process demonstrates consistent improvement, as indicated by the decreasing trend in both train and validation losses over time. Concurrently, the train and validation accuracies display an upward trajectory, albeit with fluctuations around a mean value of approximately 0.75%. It is worth noting that these fluctuations could be attributed to factors such as dataset variability, inherent noise, or model complexity.
However, the ultimate measure of performance lies in the test accuracy on the entire test set. For the latent space representations of patches 16x16 and 32x32, the obtained test accuracies are approximately 63% and 59%, respectively. These results provide insights into the model's ability to generalize and accurately classify unseen data instances. Nonetheless, further investigations are required to assess the significance of these accuracies and to compare them with alternative approaches, such as contrastive learning from visual features and multimodal contrastive learning utilizing both bathymetry and visual images.
The forthcoming comparative analysis will shed light on the relative strengths and weaknesses of the proposed methodology and enable a comprehensive evaluation of its effectiveness in capturing relevant patterns and discriminating between different bathymetry classes.

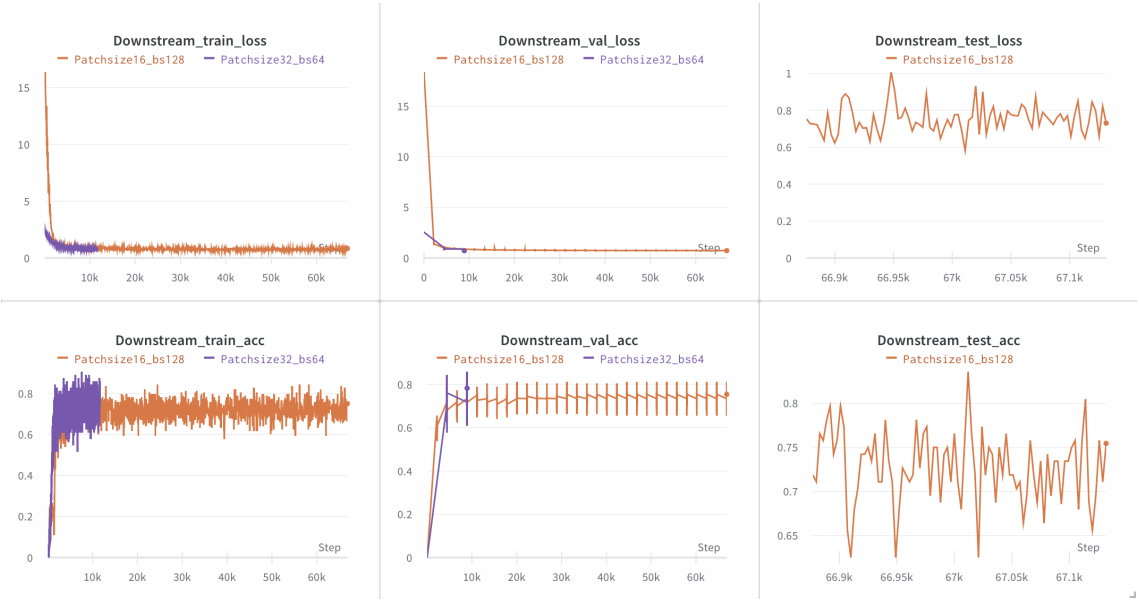



Figure 34: Logictic Regression as a downstream task of classification on feature representations from SimCLR

### 5.2.3 Visual Feature Learning

**Feature Learning**

The approach 1 related to visual feature learning outlined in Figure 20 was employed for this case. The SimCLR model was applied to visual images with dimensions of 1360x1024 pixels. The training process involved batch sizes of 256 and 512, as larger batch sizes were observed to enhance the model's training performance. The model was trained for 300 epochs using two variations of the dataset: one comprising 36,000 visual images, including both labeled and unlabeled samples, and the other consisting solely of labeled visual images.

The outcomes of the training procedure are depicted in Figure 35. Notably, the loss values exhibited a decreasing trend for both the train and validation sets, indicating an improvement in the model's ability to minimize the discrepancy between predicted and true labels. Simultaneously, the accuracy values displayed an upward trend, signifying an increasing precision in the model's predictions.

Comparing the two dataset variations, it is observed that the accuracy values for the larger dataset surpassed those of the smaller dataset by a slight margin in terms of both top-1 and top-5 accuracy metrics. The top-1 accuracy values for both datasets fluctuated around 85% for both the train and validation sets, while the top-5 accuracy values hovered around 95%.

Moving forward, the next step in the pipeline involves training a classifier on the extracted features obtained from this feature extractor. This classifier will further refine the predictions based on the encoded visual representations, potentially enhancing the model's discriminative capabilities and enabling more accurate classification outcomes.

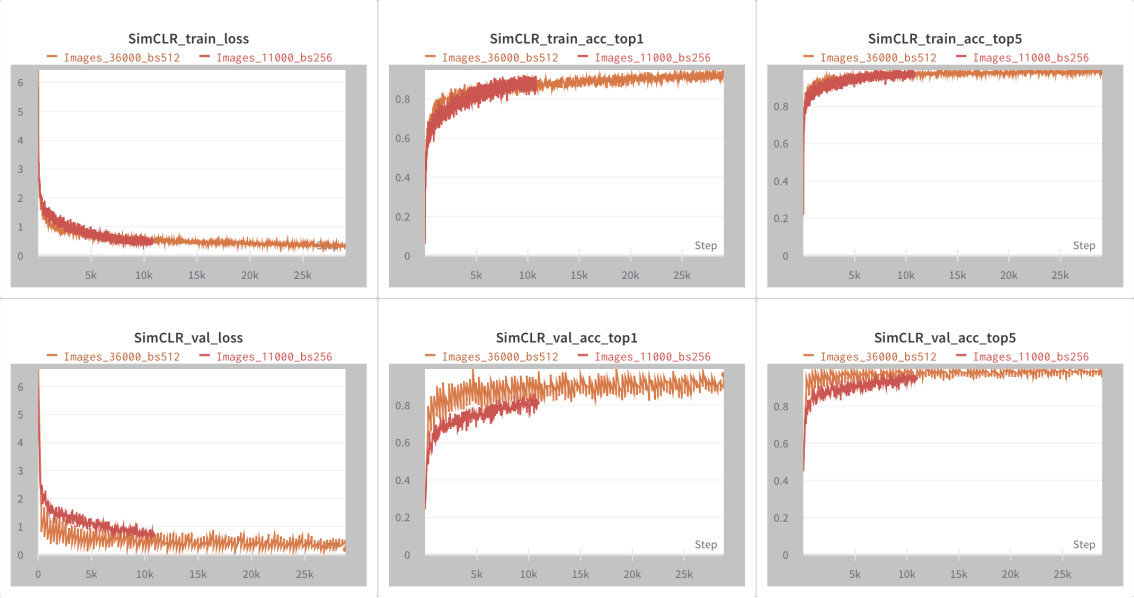



Figure 35: SimCLR for latent representation learning from visual images of size 1360x1024

### Classification

After extracting feature representations through contrastive learning for visual images, the next step involves classification using a Logistic regression approach. The training process focuses on utilizing the latent space representations derived from the SimCLR model, which was initially trained on a larger visual image dataset. Two different dataset splits are considered: a random split set and a line split set. Both splits employ a batch size of 64 during training.
The training procedure and its outcomes are illustrated in Figure 36. It is evident that the training and validation losses decrease over time for both dataset splits. However, the validation loss exhibits more significant fluctuations in the case of the line split set. On the other hand, the training accuracy steadily increases for both the training and validation sets.
The final accuracy values for the line split set are 91% for the training data and 86% for the test data, indicating a strong classification performance. These accuracy values demonstrate the model's ability to effectively classify visual images based on the extracted latent space representations.
Moving forward, the subsequent task involves incorporating multimodal learning to enable the model to learn from both visual and bathymetry data. By integrating these two modalities, the model can leverage the combined information to further enhance its classification capabilities and improve the overall performance.

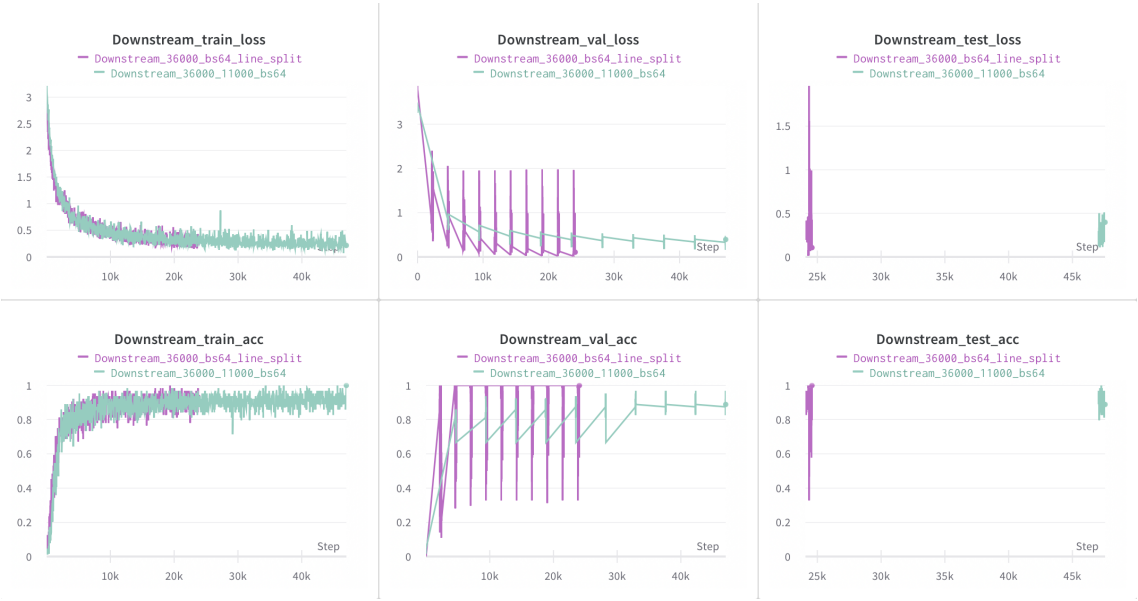



Figure 36: Logictic Regression as a downstream task of classification on feature representations from SimCLR

### 5.2.4 Multimodal learning from visual and bathymetric features

**Feature Learning - single network**

The approaches 2 and 3 outlined in Figures 21 and 22, resepctively, was employed for the following cases. For the purpose of multimodal feature learning, two distinct approaches were employed. The first approach involved training a single network capable of handling both bathymetry and optical images. In this setup, augmentations of bathymetry patches were contrasted against augmentations of visual images. Given that the two modalities possess different representations, with bathymetry patches being single-channel and smaller in size, while RGB images consist of three channels and have larger dimensions, it was necessary to transform them into a unified representation.

Two potential transformation options were considered. The first option entailed converting the visual image to grayscale and reducing its size. The second option involved transforming the bathymetry patches by expanding them into three-channel images through channel stacking. In this case, the former option of converting the visual image to grayscale was selected. The dataset provided to the model encompassed paired images, including augmented views of bathymetry patches and augmented views of visual images, along with non-paired bathymetry patches sampled from the entire bathymetry map.

The model was trained for 300 epochs, employing a batch size of 512 for patches sized 16x16 and 32x32. The results of the training procedure are depicted in Figure 37. It can be observed that the training and validation losses decreased over time, while the top-1 and top-5 accuracy values exhibited an upward trend throughout the training process. The top-5 accuracy for both the training and validation sets reached approximately 70

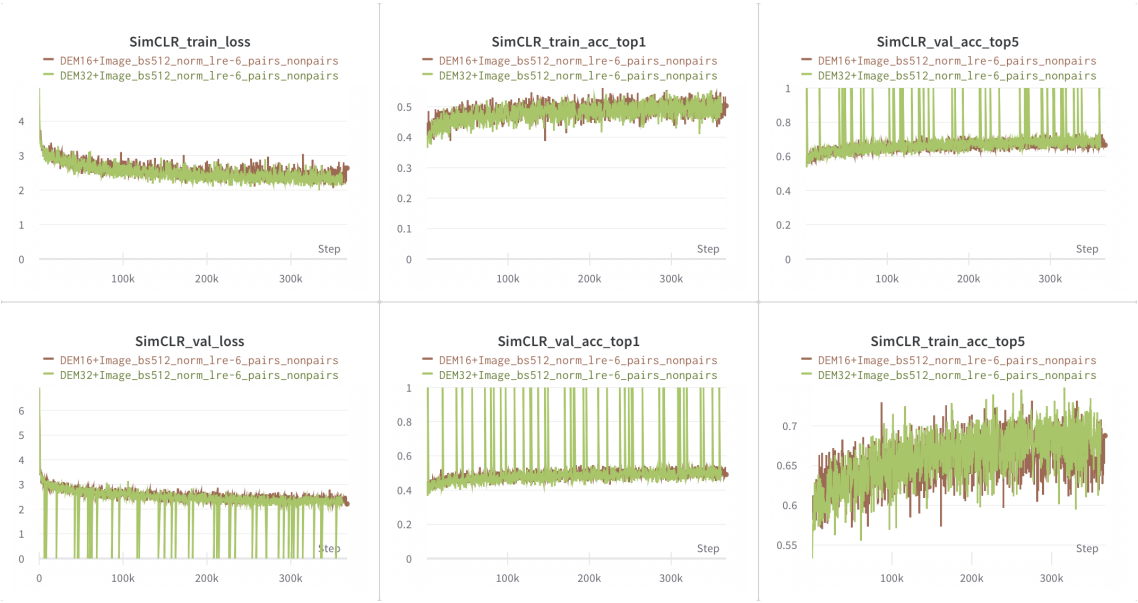



Figure 37: Single network SimCLR for latent representation learning from bathymtry and visual images

**Classification - single network**

After obtaining feature representations from bathymetry images using contrastive learning on both datasets, logistic regression is employed to classify the extracted representations of bathymetry patches. The training process for bathymetry patches of sizes 16x16 and 32x32 is presented in Figures 38 and 39, respectively, considering both the random split and line split sets.

As depicted in the figures, the training and validation losses decrease over the course of training, with the validation loss showing more pronounced fluctuations for the line split case. The final train and test accuracies for the line split set were determined to be 76% and 71%, respectively. These results indicate the model's ability to effectively classify bathymetry patch representations, achieving reasonably high accuracy levels.

Moving forward, the next step involves employing two separate networks with a shared loss function to facilitate the learning of multimodal features. This approach aims to enhance the model's capability to capture and leverage the complementary information present in both the bathymetry and optical image modalities, further improving the overall performance of the system.

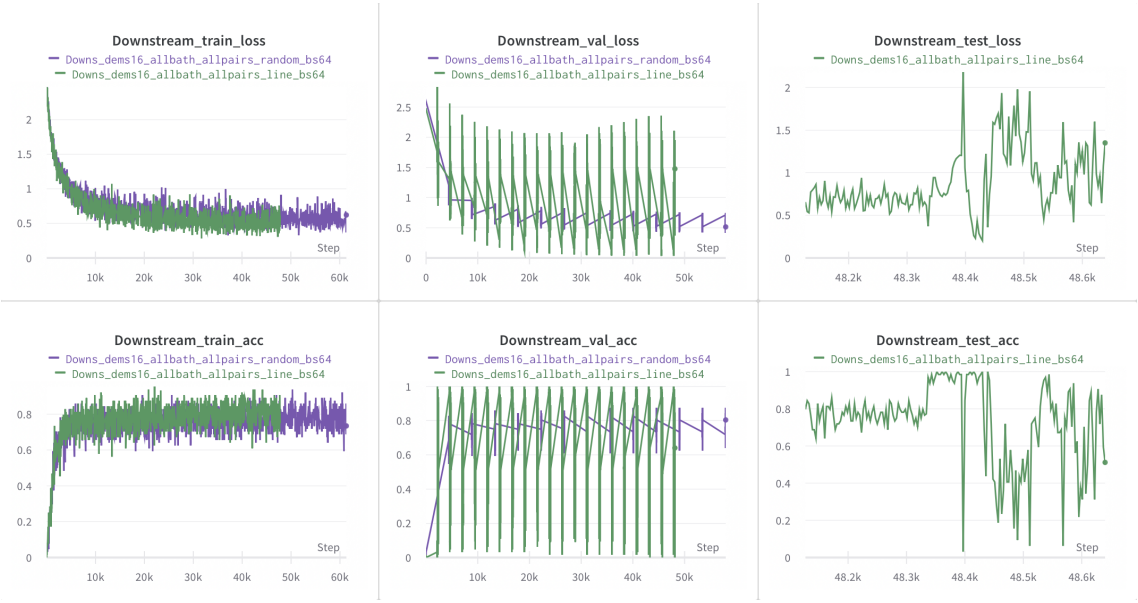



Figure 38: Logictic Regression as a downstream task of classification on feature representations from SimCLR for patches of 16x16

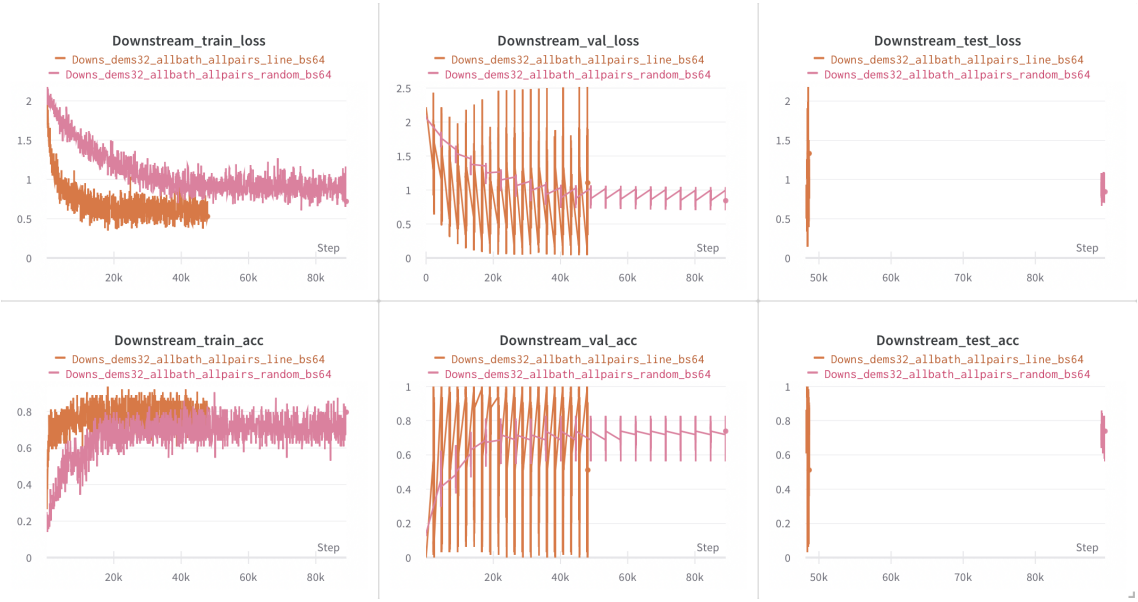



Figure 39: Logictic Regression as a downstream task of classification on feature representations from SimCLR for patches of 32x32

**Feature Learning - double network**

The approach 4 outlined in Figure 23 was employed for this case. The second approach involved utilizing separate networks for each modality and combining their losses during backpropagation. This approach eliminated the need to modify the representation of either modality, such as adjusting the number of channels or dimensions. The dataset used for training consisted solely of paired samples, consisting of corresponding bathymetry patches of sizes 16x16 or 32x32 and visual images.

The training procedure was carried out for 300 epochs with a smaller batch size of 256, which was chosen to accommodate memory limitations. The results of the training process for the bathymetry network and the visual image network are presented in Figures 40 and 41, respectively.

Analyzing the training results for the bathymetry modality, it can be observed that the train loss

gradually decreased over time. The validation loss also showed a decreasing trend but with some fluctuations. The top-1 accuracy exceeded 60% for the 32x32 bathymetry patches and reached approximately 70% for the 16x16 bathymetry patches. The validation top-1 accuracy achieved around 60% for both patch sizes. In terms of top-5 accuracy, both the training and validation sets surpassed 80%.
Examining the training results for the visual image modality, the top-5 accuracy remained around 40%. Further analysis and refinement of the model's performance on visual images will be necessary to improve its accuracy and align it with the performance achieved on the bathymetry modality.

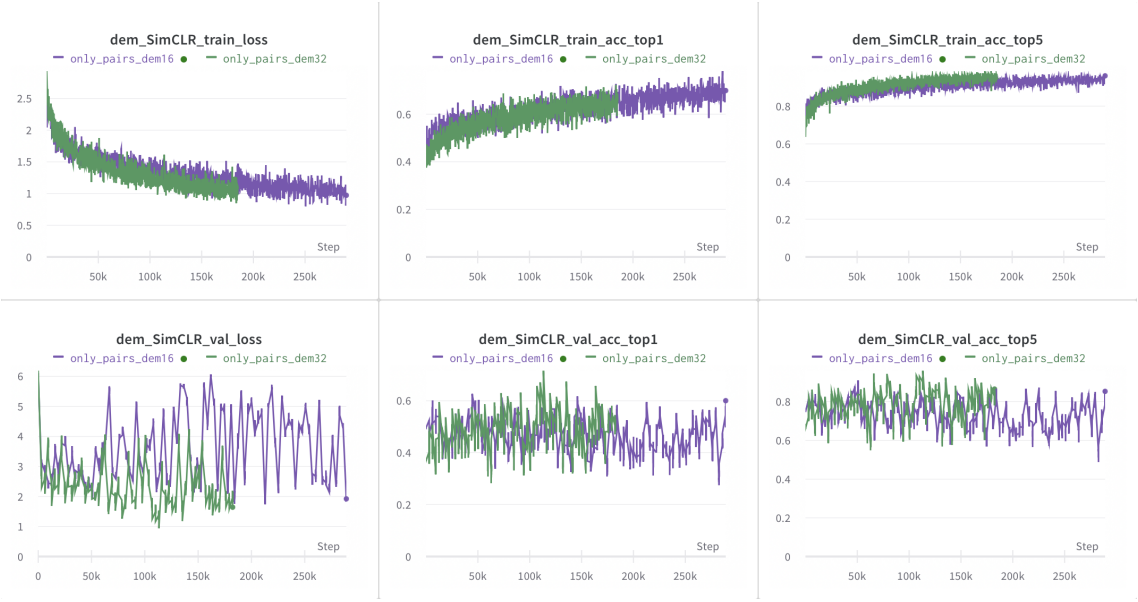



Figure 40: SimCLR with double network for latent representation learning from bathymetry patches of 16x16 and 32x32

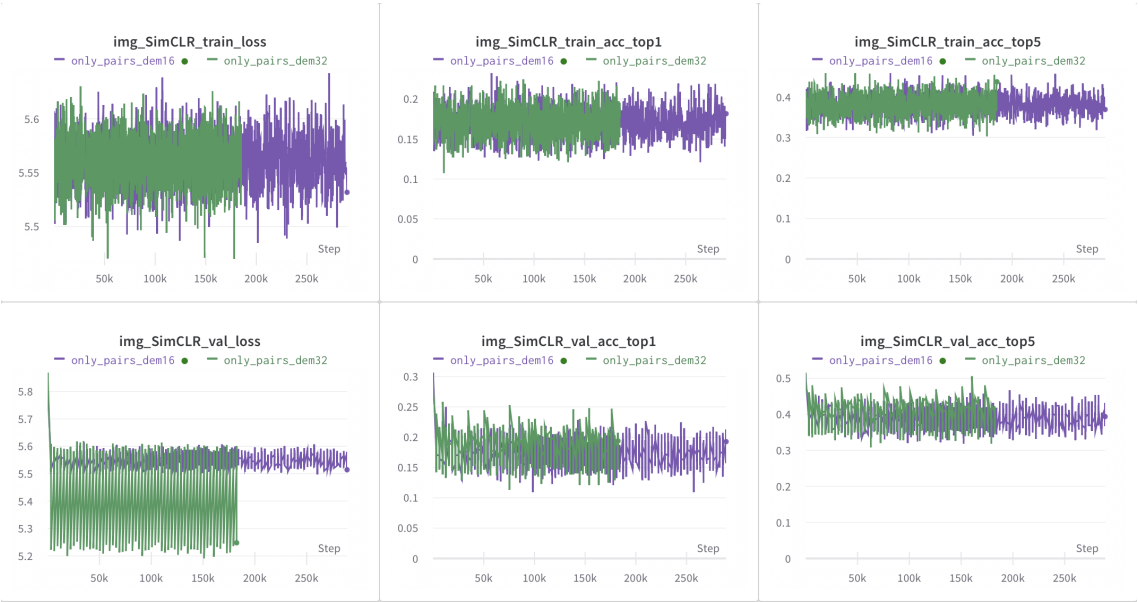



Figure 41: SimCLR with double network for latent representation learning from visual images

**Classification - double network**
Logistic regression was employed to perform a classification task aimed at identifying the labels

corresponding to image and bathymetry patch locations, utilizing the feature representations obtained through contrastive training for both bathymetry and visual images.
The training process encompassed bathymetry patches with dimensions of 16x16 and 32x32, as well as visual images, using random and line split sets. A detailed exposition of the training procedure is presented in Figures 42, 43, and 44.
Analyzing the outcomes pertaining to the 16x16 bathymetry patches, it is apparent that both the training and validation losses decreased over the course of the training iterations. Notably, the line split set exhibited relatively higher fluctuations compared to the random split set. Concurrently, the training and validation accuracies exhibited a progressive increase, culminating in final values of 76% and 71% on the line split set, respectively. Similar observations were made for the 32x32 bathymetry patches, with the classification accuracy approximating 76% for both the training and test sets on the line split set.
Turning to the classification results for the visual images, the accuracy achieved on the line split set amounted to 76% for the training set and 75% for the test set.
These findings underscore the effectiveness of feature learning through contrastive learning with further dowsntream task accurately classifying the acquired feature representations, thereby facilitating the identification of labels for the corresponding image and bathymetry patch locations. Moreover, the consistency of performance across different patch sizes and modalities demonstrates the robustness and generalizability of the feauture learning approach.

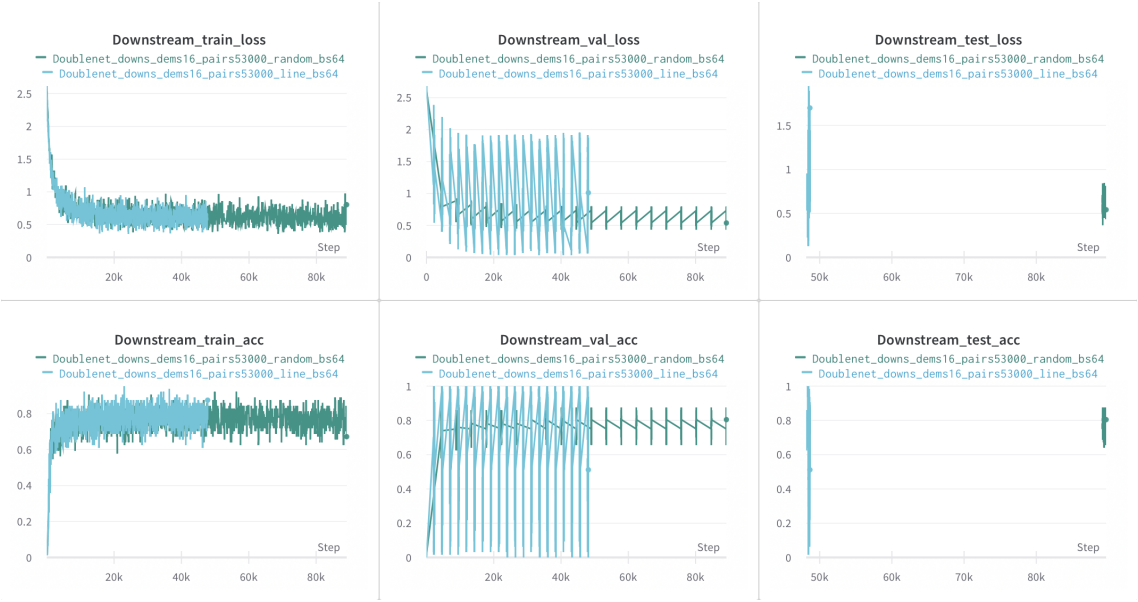



Figure 42: Logictic Regression as a downstream task of classification on feature representations from SimCLR with double network for patches of 16x16

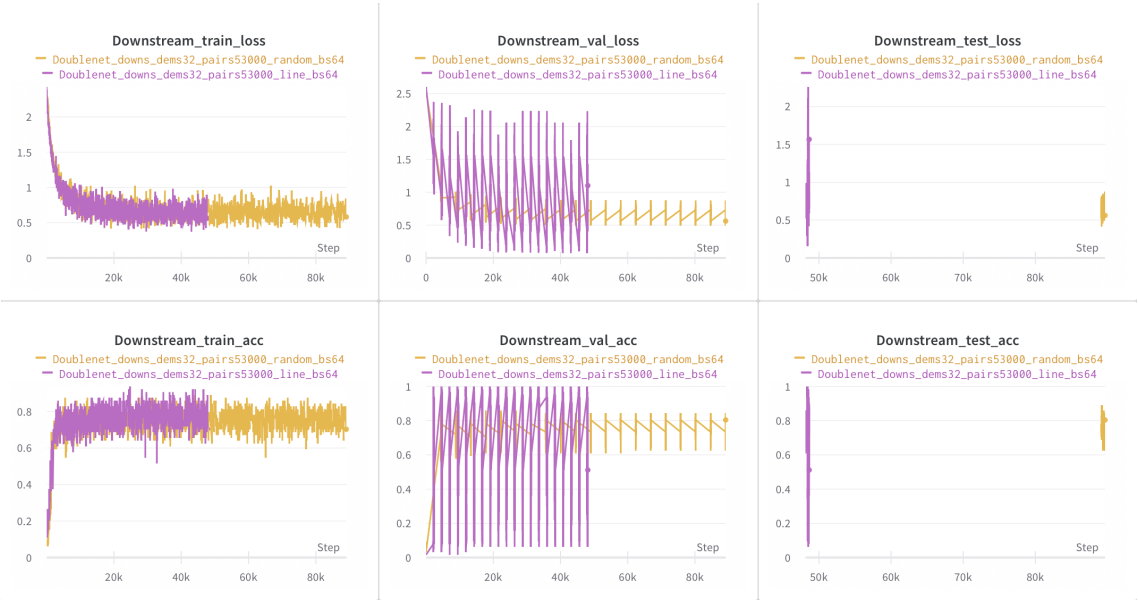



Figure 43: Logictic Regression as a downstream task of classification on feature representations from SimCLR with double network for patches of 32x32

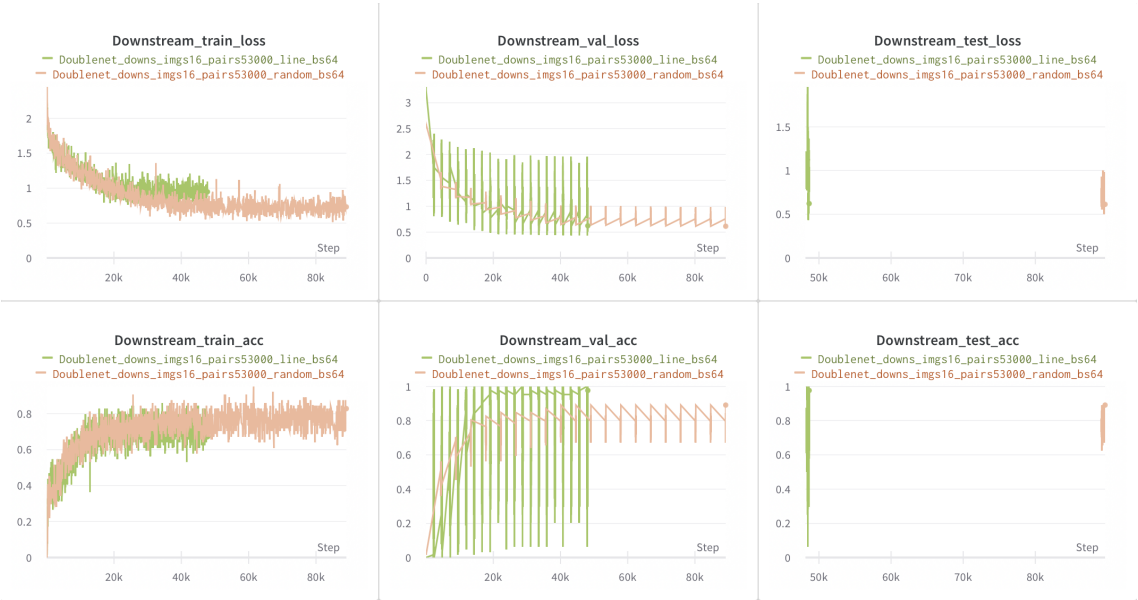



Figure 44: Logictic Regression as a downstream task of classification on feature representations from SimCLR with double network for visual images

# 6 Discussion

The final results depicting the performance of different approaches are presented in Table 1. These performance metrics are based on the line split of the labeled dataset, where the straight line was utilized for training (approximately 40% of the labeled data) and the zigzag path was used for testing (about 60% of the labeled data). The table reveals that conducting contrastive learning on a large amount of bathymetry data and subsequently performing classification using the extracted features yielded test accuracy rates of approximately 59% and 63% for patch sizes 16x16 and 32x32, respectively. This lower performance can be attributed to the lower resolution of the bathymetry data.
In contrast, performing the same contrastive learning on visual images resulted in a test accuracy performance of over 86%. This signifies the superior performance of visual images, which possess high angular resolution and offer detailed spatial information about the observed scene.
Regarding the multimodal learning approach, where visual images were contrasted with bathymetry patches within a single network, it yielded accuracies of about 71% and 72% for patch sizes 16x16 and 32x32, respectively. Furthermore, employing separate networks for visual and bathymetry data, with a shared loss function to preserve their representations, achieved accuracies of approximately over 71%. These results highlight the benefits of multimodal learning, as it enables one modality with extensive coverage capabilities to learn rich features from the other modality, which may have a narrower scope. In this context, synchronous learning of the feature space of visual images, with their high angular resolution and detailed spatial information, and bathymetry patches, providing valuable depth information about the underwater topography and seafloor characteristics despite their lower resolution, proved to be advantageous.
Thus, the utilization of multimodal learning with the contrastive learning technique demonstrated its feasibility and effectiveness in this study.

Table 1: Table of performance results of training classifier on feauture representations from contrastive learning.

| **SimCLR model** | **Approach** | **Split** | **Train accuracy (%)** | **Test accuracy (%)** |
|---|---|---|---|---|
| Bathymetry 16 | 1 | Line | 77.21 | 63.97 |
| Bathymetry 32 | 1 | Line | 65.22 | 59.79 |
| Visual Images | 1 | Line | 91.03 | 86.10 |
| Single network bathymetry 16 and image pairs | 2,3 | Line | 76.83 | 71.12 |
| Single network bathymetry 32 and image pairs | 2,3 | Line | 77.84 | 71.89 |
| Double network bathymetry 16 and image pairs on bathymetry | 4 | Line | 76.36 | 71.59 |
| Double network bathymetry 32 and image pairs on bathymetry | 4 | Line | 76.25 | 71.23 |
| Double network bathymetry 16 and image pairs on images | 4 | Line | 76.34 | 74.08 |

# 7 Conclusion

In conclusion, this thesis work aimed to explore multimodal learning techniques involving remotely sensed and visual data in the context of habitat classification. Gathering visual imagery of large underwater areas is a resource-intensive task, while bathymetry data obtained from multibeam sonar devices provides a more easily accessible representation of the underwater terrain. Combining these modalities allows for efficient and reliable habitat classification. The limited availability of underwater visual imagery can serve as ground truth, while the widespread availability of bathymetry data can be used to estimate and predict habitat classes based on the imagery data. By leveraging both modalities, they can complement each other and contribute to a common task and goal.
In this master's thesis work, self-supervised multimodal learning techniques, specifically contrastive learning, were investigated, implemented, and tested on terrestrial and underwater datasets. The results demonstrated the mutual benefit of each modality. Multimodal learning enhanced the performance of models in predicting bathymetry data by leveraging the contrastive learning of features from abundant but low-quality bathymetry data and scarce but rich visual data.

## 7.1 Future work

In future work, it is anticipated that the predictions of the habitat classes will be visualized across the entire bathymetry map. Another future research in this domain entails the incorporation of uncertainty models into the existing approaches to facilitate uncertainty estimation in predictions. This would enable the generation of efficient sampling trajectory plans for autonomous underwater vehicle (AUV) surveys. By identifying areas with higher uncertainty, subsequent AUV surveys can focus on collecting data from these regions, thereby improving the model's performance and enhancing habitat classification. This approach also has the potential to reduce costs associated with data collection.
Furthermore, additional work extends to exploring additional self-supervised multimodal learning approaches that can be applied to underwater datasets. The aim is to identify the most effective models that achieve high scores on evaluation metrics such as accuracy, precision, and recall. Continued research and development in this field will contribute to advancing the understanding and application of multimodal learning techniques in underwater data analysis.

# Bibliography


Abdulazizov, Shakhboz, Thomas Trappenberg and Scott C Lowe (n.d.). 'An Autoencoder Model of Bathymetry and Multibeam Echosound Backscatter'. In: ().

Ahsan, Nasir, Stefan B Williams and Oscar Pizarro (2012). 'Robust broad-scale benthic habitat mapping when training data is scarce'. In: *2012 Oceans-Yeosu*. IEEE, pp. 1–10.

*ALOS Global Digital Surface Model "ALOS World 3D - 30m (AW3D30)"* (2023). URL: https://www.eorc.jaxa.jp/ALOS/en/index_e.htm.

Bender, Asher, Stefan B Williams and Oscar Pizarro (2012). 'Classification with probabilistic targets'. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1780–1786.

Brown, Craig J et al. (2011). 'Benthic habitat mapping: A review of progress towards improved understanding of the spatial ecology of the seafloor using acoustic techniques'. In: *Estuarine, Coastal and Shelf Science* 92.3, pp. 502–520.

Castillo-Navarro, Javiera et al. (2022). 'Semi-Supervised Semantic Segmentation in Earth Observation: The MiniFrance suite, dataset analysis and multi-task network study'. In: *Machine Learning* 111.9, pp. 3125–3160.

Chen, Ting et al. (2020). 'A simple framework for contrastive learning of visual representations'. In: *International conference on machine learning*. PMLR, pp. 1597–1607.

Dosovitskiy, Alexey et al. (2020). 'An image is worth 16x16 words: Transformers for image recognition at scale'. In: *arXiv preprint arXiv:2010.11929*.

Giurgi, Dănuţ-Vasile et al. (2022). 'Real-time road detection implementation of UNet architecture for autonomous driving'. In: *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. IEEE, pp. 1–5.

Goyal, Priya et al. (2021). 'Self-supervised pretraining of visual features in the wild'. In: *arXiv preprint arXiv:2103.01988*.

He, Kaiming et al. (2016). 'Deep residual learning for image recognition'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Heidler, Konrad et al. (2021). 'Self-supervised audiovisual representation learning for remote sensing data'. In: *arXiv preprint arXiv:2108.00688*.

*How Multibeam Sonar Works* (2009). URL: https://oceanexplorer.noaa.gov/explorations/09bermuda/background/multibeam/multibeam.html.

Le-Khac, Phuc H, Graham Healy and Alan F Smeaton (2020). 'Contrastive representation learning: A framework and review'. In: *Ieee Access* 8, pp. 193907–193934.

Le, Quoc V (2013). 'Building high-level features using large scale unsupervised learning'. In: *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, pp. 8595–8598.

Li, Wenyuan, Hao Chen and Zhenwei Shi (2021). 'Semantic segmentation of remote sensing images with self-supervised multitask representation learning'. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, pp. 6438–6450.

Liang, Zongbao et al. (2021). 'UAV-based cross-view geo-localization fusion spatial attention mechanism and Netvlad'. In: *2021 7th International Conference on Systems and Informatics (ICSAI)*. IEEE, pp. 1–6.

*National Land Cover Database* (2023). URL: https://www.naturvardsverket.se/en/services-and-permits/maps-and-map-services/national-land-cover-database?_t_hit.id=Boilerplate_Episerver_Features_EpiserverFind_Models_EpiserverFindDocument2F8461_enamp;_t_q=national+land+coveramp;_t_id=JZoPwJYqToSxW92mfsv1cwamp;_t_tags=siteid3A69c7ea6e-2b02-4832-8c8c-31da973f12f12Clanguage3Aen.

Rakhlin, Alexander, Alex Davydow and Sergey Nikolenko (2018). 'Land cover classification from satellite imagery with u-net and lovász-softmax loss'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 262–266.

Rao, Dushyant, Mark De Deuge, Navid Nourani–Vatani et al. (2017). 'Multimodal learning and inference from visual and remotely sensed data'. In: *The International Journal of Robotics Research* 36.1, pp. 24–43.

Rao, Dushyant, Mark De Deuge, Navid Nourani-Vatani et al. (2014). 'Multimodal learning for autonomous underwater vehicles from visual and bathymetric data'. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 3819–3825.

Ronneberger, Olaf, Philipp Fischer and Thomas Brox (2015). 'U-net: Convolutional networks for biomedical image segmentation'. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, pp. 234–241.

Shields, Jackson, Oscar Pizarro and Stefan B Williams (2020). 'Towards adaptive benthic habitat mapping'. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 9263–9270.

— (2021). 'Feature space exploration for planning initial benthic AUV surveys'. In: *arXiv preprint arXiv:2105.11598*.

Shrivastava, Ashish et al. (2017). 'Learning from simulated and unsupervised images through adversarial training'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2107–2116.

Spinoccia, M (2011). 'Bathymetry grids of south east tasmania shelf'. In: *Geosciences Australia*.

Wang, Yi et al. (2022). 'Self-supervised learning in remote sensing: A review'. In: *arXiv preprint arXiv:2206.13188*.

Williams, Stefan B et al. (2010). 'AUV benthic habitat mapping in south eastern Tasmania'. In: *Field and Service Robotics*. Springer, pp. 275–284.

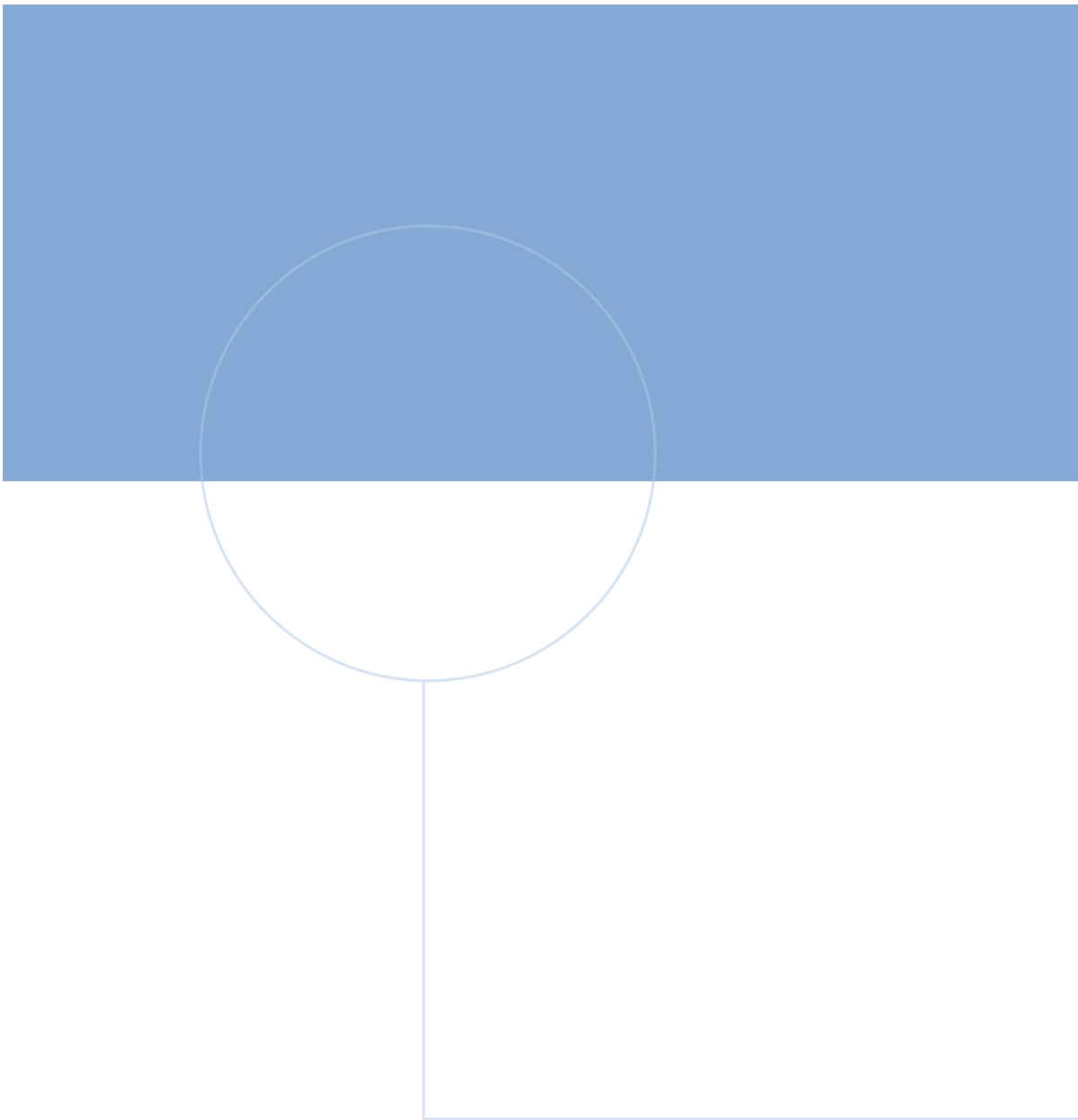
Yamada, Takaki et al. (2022). 'Guiding labelling effort for efficient learning with georeferenced images'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

# Appendix

Codes related to this thesis are included in the Github repository linked below.

## Github repository link

- https://github.com/AzamatKaibaldiyev/Contrastive_learning_Underwater



NTNU

Norwegian University of
Science and Technology