Masteroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk

Martin Lund Haug

Applying active learning techniques in machine learning to minimize labeling effort

Masteroppgave i kybernetikk og robotikk
Veileder: Annette Stahl
Medveileder: Aya Saad
Juni 2021

NTNU
Norwegian University of
Science and Technology

Martin Lund Haug

Applying active learning techniques in machine learning to minimize labeling effort

Masteroppgave i kybernetikk og robotikk
Veileder: Annette Stahl
Medveileder: Aya Saad
Juni 2021

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk

NTNU
Norwegian University of
Science and Technology

Abstract

The most prominent machine learning (ML) methods for classification rely heavily on a massive amount of labeled data to create and train neural network classifier models that perform their tasks accurately. With the complex structure of planktonic species and an immense amount of data captured from autonomous underwater vehicles (AUVs), a large burden is placed on the domain experts for plankton taxa labeling.

Active Learning (AL) is an ML paradigm that reduces this manual effort by proposing algorithms that support the construction of the training datasets, thus enlarging the sets while minimizing human involvement. To build the training set, AL methods apply heuristics to select a subset of images, i.e., samples, from the entire data. The applied AL algorithm should select samples that capture the common statistical patterns or feature space and are likely to include all the information needed for the training and the learning processes. In addition, the algorithm should prioritize samples that are likely belonging to multiple classes, i.e., having close inter-class boundaries, and might lead to model confusion. Many of the current AL approaches fail to incorporate both types of samples representing the statistical pattern and the samples in which the particular machine learning model is uncertain about. Inspired by these limitations, this thesis presents a novel framework that combines these two types of sampling to utilize the full data distribution, prevent redundant sampling from correlated queries, and fine-tune the inter-class decision boundary.

The results from extensive experiments on the proposed framework and methods from the AL literature show that several of the methods lack robustness to different experimental conditions. However, the proposed hybrid framework proves to be robust and accurate on complex active learning tasks and competitive with other active learning strategies under various experimental conditions. The thesis further shows that the employment of a data augmentation module enhances the overall classification performance and in particular can benefit the sampling strategy in an AL framework.

Contents

List of Tables

List of Figures

Abbreviations

Abbreviation	Description
AL	Active learning
RBS	Random Benchmark Sampling
FN(R)	False negative (rate)
FP(R)	False positive (rate)
LC	Least confidence
ML	Machine learning
MS	Margin sampling
TN(R)	True negative (rate)
TP(R)	True positive (rate)
ANN	Artificial neural network
BNN	Bayesian neural network
CNN	Convolutional neural network
SVM	Support vector machine
ReLU	Rectified linear unit
ResNet	Residual neural network
FC	Fully connected
AUV	Autonomous underwater vehicle
GPU	Graphical processing unit
DAL	Deep active learning
MC	Monte Carlo
UQ	Uncertainty Quantification

Preface

This master's thesis is submitted as a part of the requirements for the master's degree at the Department of Engineering Cybernetics at the Norwegian University of Science and Technology. The work with this thesis was done as a part of the AILARON[1] project and is a continuation of the specialization project conducted by the undersigned during the fall semester of 2020. Based on the work conducted in this thesis, I was invited to present my findings at the "13th International Conference on Digital Image Processing" held in Singapore on 22nd-25th May 2021. The title of my presentation was "A combined informative and representative active learning approach for plankton taxa labeling" and was presented under the section "Digital Image Processing and Methods". An extension of the presented work has been submitted at the "13th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles" and is at the time of writing under review for acceptance.

I would like to thank my supervisors Aya Saad and Annette Stahl, from the Department of Engineering Cybernetics for their support and guidance throughout this project.

Martin Lund Haug
07/06/2021

[1] AILARON is a multidisciplinary project seeking knowledge of the plankton species and their distributions. This research is funded by the RCN FRINATEK IKTPLUSS program (project number 262741) and supported by NTNU AMOS.

Chapter 1

Introduction

Planktonic species are critically important to the oceanic ecological structure as they are the basis of the aquatic food web. Hence, by studying temporal variations in plankton taxa distribution, one can achieve a proxy for the development of the oceanic ecosystem.

Progress in the development of autonomous underwater vehicles (AUVs) and robotic visual sensing enables the possibility of capturing large amounts of planktonic image data. Further, Convolutional Neural Network (CNN) models have proved competent at solving computer vision problems in the supervised Machine Learning (ML) paradigm. Embedding CNN models into AUVs enables the identification of plankton taxa distributions in-situ. However, modern CNNs require an immense amount of pre-classified input to achieve satisfactory classification performance. Since plankton biomass appears in many different species, forms, and stages depending on the geographical environment and season, pre-classified training data has to be constructed for each different geographical environment, season, and image-acquiring system. Consequently, much effort is needed for the manual plankton taxa labeling that requires domain expertise, i.e., biologists, to identify the complex structure of planktonic organisms. Active Learning (AL) is a semi-supervised machine learning approach that aims at mitigating this burden placed on domain experts. By leveraging samples with a high amount of information, it is possible to sufficiently capture the data distribution of a full dataset with only a fraction of the samples, hence minimizing the manual labeling effort.

1.1 Aim of study

The overall aim of the study conducted throughout this thesis has been to develop a method to minimize the manual effort on plankton taxa labeling. An intermediate objective has been to gain knowledge on research in the area of active learning and identify gaps in existing methods proposed in the literature. Additionally, research on planktonic species and their classification has been relevant to adopt AL strategies to the planktonic domain.

1.2 Research questions

This section will provide some insight into the research questions this thesis is intended to answer. These questions are considering relevant research in the field of AL, considerations for implementation of AL, and prominent challenges in the field. Hopefully, these questions would encourage further reading and give insight into the ideas and challenges of AL.

- **What is the current standing of research in the field of AL and in particular for the planktonic domain?** To develop and adopt AL methods for plankton taxa labeling, it is essential to gain an overview of previous work and research in the field. This can be obtained through experiments on recent modes and methods from the AL literature combined with an analysis of their advantages and limitations.
- **What are considerations and challenges related to the implementation of AL?** This research question intends to give a better overview of the considerations needed when implementing an AL framework. In particular, it aims to investigate the connection between the employed dataset, the AL strategy and the number of queries in each round, and examine how it is related to the performance of AL. The research question will also identify challenges and limitations related to AL for deep learning.
- **How can the challenges in current AL approaches be mitigated?** This research question concerns how the obtained knowledge on modes and methods in AL can be used to mitigate gaps and challenges identified in the AL literature. In other words, how can a novel framework be designed to mitigate the challenges faced by other AL approaches in the literature?

By addressing these questions, the thesis aims to give a good understanding of active learning and present its current position in the literature together with current gaps and challenges. Further, the research and experiments conducted throughout this thesis

are intended to provide additional aspects of the methods proposed in the literature including their strengths and weaknesses. Finally, by focusing on the aforementioned challenges, the framework proposed in this thesis intends to provide an accurate and robust AL strategy for adoption to the planktonic domain.

1.3 Contributions

The contributions of this thesis are threefold.

- A thorough research is conducted on modes and methods of active learning to provide knowledge on its current standing and related challenges.
- A novel hybrid framework for active learning that proves to be suitable for the planktonic domain and in particular for the AILARON SilCam dataset is proposed. The proposed hybrid framework is designed to mitigate challenges identified in the AL literature. A paper presenting the proposed framework was submitted and accepted at the *"13th International Conference on Digital Image Processing"* [2]. Furthermore, an extension of the work focusing on the adoption to the planktonic domain is currently under review at the *"13th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles"* [1].
- Several different methods covering the broad categories of deep active learning strategies are compared on both a benchmark dataset and three complex datasets from the planktonic domain. The results are analyzed and relevant challenges and considerations of active learning are discussed.

The aforementioned results and discussion are intended to assist the reader in choosing the right strategy and parameters when implementing AL to minimize labeling effort and speed up the construction of training datasets.

1.4 Outline

The rest of the thesis is organized as follows. Chapter 2 presents relevant background knowledge on topics of machine learning with an emphasis on image classification and active learning. The background presented is considered a precondition for a proper understanding of the rest of the thesis and the experiments presented. The related work presented in chapter 3 considers previous research on minimizing labeling effort in the planktonic domain and general AL strategies related to the proposed hybrid framework. The datasets employed in the experiments in this thesis are presented in

chapter 4. Moreover, chapter 5 describes the set-up and methodology for the hybrid active learning method proposed in this thesis, the chapter covers the framework, employed image classifiers, and relevant implementation details. Furthermore, results from the experiments are reported in chapter 6 followed by short analysis and summaries. The discussion in chapter 7 aims to answer the research question presented in the introduction by linking up the background material with an analysis of the experimental results. The findings of the thesis are summarized in chapter 8, emphasizing the most important results. Lastly, interesting future research directions are presented in chapter 9.

Chapter 2

Background

The background chapter presents relevant knowledge in the field of machine learning and its sub-domains computer vision and active learning. The concepts presented are considered relevant for a further understanding of the work presented later in the thesis. To begin with, a general introduction to ML is presented, then CNN and its components are introduced due to their are high relevance for plankton taxa classification. Finally, active learning and deep active learning are thoroughly presented with their modes and methods.

2.1 Machine Learning

Machine Learning (ML) is a branch of data science where computers are allowed to learn from data and make classifications or predictions, based on learned attributes when presented with new data. ML is inspired by how humans extract and label patterns to learn. It has a wide range of applications and is increasingly adopted to new areas, however, this thesis will mainly cover the application of computer vision and image classification. The field of computer vision is considered one of the very successful applications of ML, where credit should be given to the development of convolutional neural networks (CNNs) which leverages high computational power and large amounts of image data. This category of artificial neural networks (ANNs) will be further elaborated in section 2.2.1.

For a general ML model, one can classify images in a *supervised* or *unsupervised* way. The former trains a model to map sets of image features $\mathcal{X} = \{x_1, .., x_n\}$ to a given label $\mathcal{Y}$, whereas the latter is only concerned with the sets of features $\mathcal{X} = \{x_1, .., x_n\}$ to

discover patterns and commonalities between them. Further, when only a small amount of labeled data is available, a third way, *semi-supervised* learning is employed. Semi-supervised learning is midway between the two aforementioned categories and is the domain of Active Learning which is the topic for this thesis. Furthermore, while there exist multiple machine learning strategies for image classification such as decision trees, K-nearest neighbors, and support vector machines (SVM), this thesis will mainly concern with the field of deep learning, and from here on the terms machine learning and deep learning will be used interchangeably. However, some of the alternative ML strategies will be mentioned again as a part of the classical active learning methods described in section 2.4.

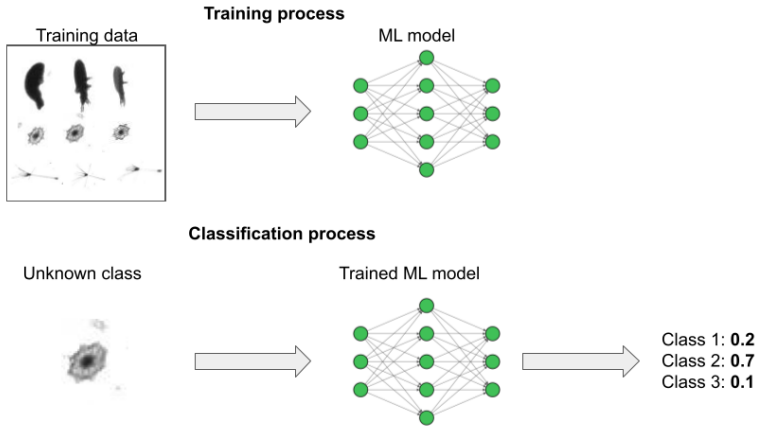


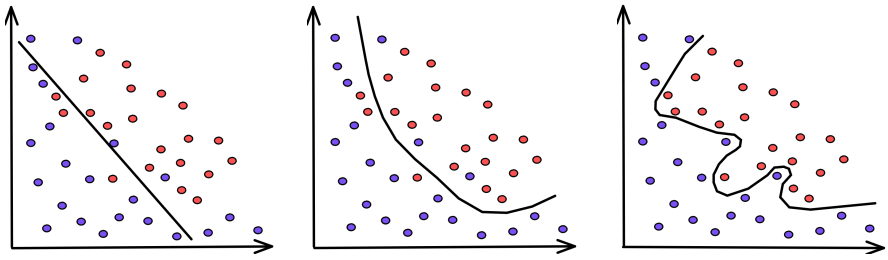
Figure 2.1: Training and classification process of an ML model.

As illustrated in figure 2.1, the general supervised and semi-supervised machine learning models do usually have two modes: training and classification. Common between the modes is that they present the machine learning model with a set of features, i.e training data, and feed it through the layers in the network to extract relevant image features. As will be further elaborated in the coming sections, the classification mode outputs a probability distribution based on the extracted image features. This can be observed in figure 2.1 where the learning model is predicting 'Class 2' for the input image. However, while the classification mode only makes a prediction, the training mode compares the prediction with the true label to calculate

how far off the prediction is. This sum, also referred to as the loss, is then propagated back through the network for the model to learn. This concept of learning through backpropagation is further elaborated in section 2.2.1 and 5.2.

Model fitting

In the classification and training process of a machine learning model, as described in the previous section, *training error* and *test error*, are two central concepts for validation of the model performance. The former represents the classification error, i.e number of wrong predictions, on the dataset the model is trained on, whereas the latter represents the classification error on a held-out dataset not seen by the model. Although the training error is helpful to see how well the model is extracting features from a given dataset, it can not be used to confidently evaluate the model performance. For this purpose, a held-out dataset, i.e validation set, is used to see how well the model can transfer its learning to new data, that is, how well it is *generalizing*. When a learning model pays little attention to the features in the training data, as seen in figure 2.2a, it is typically *under-fitting* the underlying data distribution and will typically lead to both low training and validation accuracy. An under-fitted model may suggest that a too sparse model is employed, hence not able to capture complex features of the dataset. On the other hand, a model that is *over-fitting* the underlying data distribution will have paid too much attention to the noise in the training data and will not generalize well on new, unseen data. Hence, this will typically lead to high training accuracy and low validation accuracy and indicate a too dense learning model. An illustration of over-fitting can be seen in figure 2.2c. However, a model that can trade-off between the training and validation accuracy, that is, only extract the most important patterns from the underlying data distribution and disregard any noisy data, will perform well both concerning training and validation accuracy. This concept of well-fitting the underlying data distribution is illustrated in figure 2.2b.



(a) Concept of under-fitting. (b) Concept of well-fitted model. (c) Concept of over-fitting.

Figure 2.2: Decision boundary on different levels of model fitting.

2.2 Image Classification

Computer Vision is a field of machine learning that trains computers to extract information from images to interpret the visual world. It is an important research direction for the automation of manual processes such as image classification and is resultingly applicable for industrial automation and autonomous vehicles. Computer Vision has been a research topic for many decades, however, progress in the field of deep learning and increased availability of computational power has sped up the performance and range of applications. The following section will give a brief introduction to the field of image classification and convolutional neural networks.

2.2.1 Convolutional Neural Network

Convolutional neural networks (CNNs) are a sub-class of artificial neural networks (ANNs) containing convolutional layers and are a major reason for the progress in the field of computer vision. The convolutional layers extract features to enable encoding of the images into lower-dimensional feature vectors These convolutional layers can also be regarded as learnable filters that improve their feature extraction to only extract the most relevant features so the neural network can correctly classify the input. A CNN consists of multiple layers which will be introduced and briefly described in this section.

For image processing, modern convolutional neural networks and GPUs leverage the concept of *tensors.* A tensor is a mathematical object that works as a generalization of n-dimensional arrays. For instance, a scalar is a zero-dimensional tensor whereas a vector is a one-dimensional tensor. A tensor can have all sorts of dimensions, however, for image processing, they most often have three or four. As

seen in figure 2.3, a discretized image is commonly represented as a matrix of pixel values, hence a two-dimensional tensor can represent a grayscale image with the shape [Height, Width]. However, color images need one tensor per color dimension. That is, an image represented with the red, green, and blue (RGB) color model would need one frame per color representing the strength of the color at each particular pixel. For instance, a white spot in an 8-bit RGB image is represented as [255,255,255].

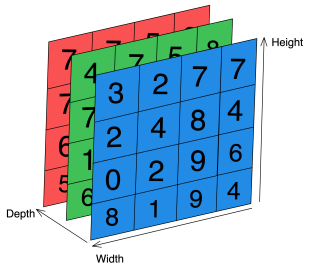


Figure 2.3: An RGB image represented by a tensor with shape [4,4,3].

These frames representing the color intensities are known as channels, or color depth, and the corresponding tensor for an RGB image can be summarized as [Height, Width, Channels] as illustrated in figure 2.3. Furthermore, when processing batches of images, all images in a batch are put together in one tensor. This adds one additional dimension, the batch size, to the tensor. Consequently, when processing images in a convolutional neural network, it is common to have tensors with the dimensions *[Batch size, Height, Width, Channels]* where all the images in the tensor need to have equal dimensions. Hence, before feeding the input image to the convolutional layer in the CNN as illustrated in figure 2.4, it needs to be converted to a tensor as illustrated in figure 2.3.

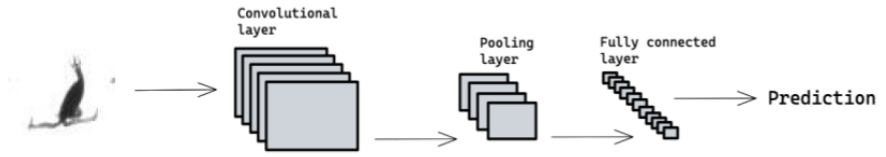


Figure 2.4: A sparse CNN consisting of a convolutionial layer, a pooling layer and a fully connected layer. To create a probability distribution for the output, a softmax layer is usually added to the fully connected layer.

The first part of a CNN is the convolutional layer. Given two functions $\mathcal{F}$ and $\mathcal{G}$, a convolution $\mathcal{F} * \mathcal{G} = \mathcal{H}$, express how the shape of one variable, $\mathcal{F}$, is modified by the other, $\mathcal{G}$. In a convolutional layer, this operation is typically two-dimensional where the first variable is an image of pixel values, i.e a tensor, and the second variable is a two-dimensional filter, as illustrated in figure 2.5. The discrete two-dimensional convolution is formally described as

$$(f * g)[x, y] = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f[x, y] g[x - m, y - n] \tag{2.1}$$

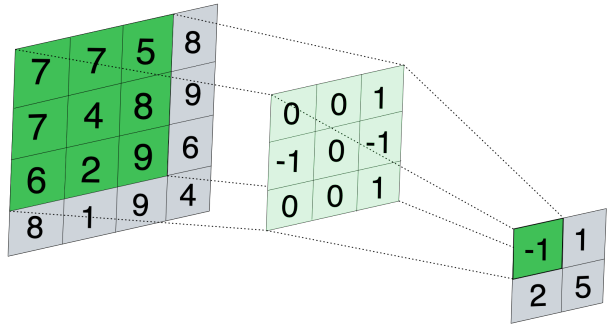


Figure 2.5: The convolutional layer. A filter (MIDDLE) is sliding over the input matrix (LEFT) to extract important features from the input.

The employed filter, often referred to as a *neuron or kernel* in the CNN literature, slides over the input image to create a feature map of a local region in the input image, as illustrated in figure 2.5. This local region, colored green in figure 2.5, is known as the *receptive field*. The size of the receptive field decides the size of the area from which the filter considers information. Based on the numerical values in the filter, referred to as *weights* in ML terminology, such filters can be handcrafted to detect vertical lines, corners, or edges in an image. With convolutional layers in a neural network, these filters are, instead of being handcrafted, learning which features to extract from a given image. Based on the output from the filter, an *activation function* decides whether to bring forward the extracted features or not. The most commonly employed activation function is *ReLU* (Rectified Linear Unit), which apply the function

$$f(x) = max(0, x) \tag{2.2}$$

to the input. In general, it forwards the value of the previous layer if the input resembles the shape the filter is representing, that is, if the values in the receptive field

correlates with the filter. The main purpose of the activation function is to combine the linear summations from the filters into a non-linear output, enabling the network to approximate non-linear inputs. Further, by combining multiple filters in parallel for a given input, as illustrated in figure 2.4, each filter can extract a specific feature, and by stacking multiple convolutional layers, the network is enabled to gradually build an understanding of the input image. This is achieved by initially extracting simple lines, corners, and edges, combine these to form shapes, and then again combine these shapes to extract domain-related objects from the input image. This process is highly flexible with regard to the input and by stacking a lot of layers, the classification accuracy of the network can become very high and compete with humans for image classification tasks [11].

Another distinctive part of CNNs is the *pooling layer*. As can be observed in figure 2.4, the pooling layer is employed after and in between the convolutional layers in the CNN. Their task is to downsample the dimension of the input to reduce the number of parameters employed and make the model invariant to local translation. This enables the classifier to recognize an object even though its position in the image is shifted compared to the training data. The dimensionality reduction is realized by sliding a pooling filter over the input and only pass on the largest value in the area (max pooling), illustrated in figure 2.6, or the average value in the area (average pooling).

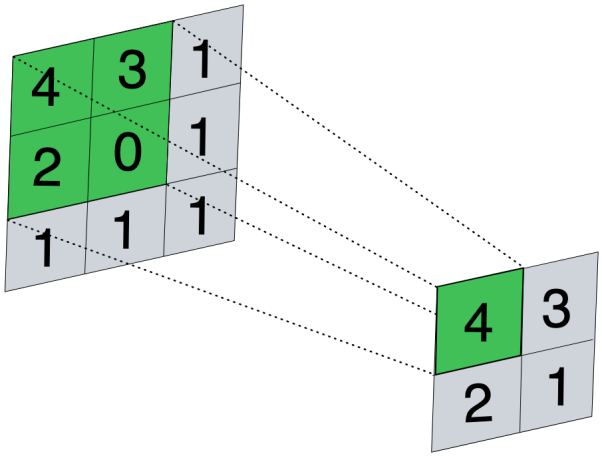


Figure 2.6: 2x2 max pooling on a 3x3 input image.

The last layer in the CNN is termed the *fully connected (FC) layer*. To bring

forward the most prominent features from the convolution and pooling process and optimize the class scores, the three-dimensional tensors are *flattened* and fed into the FC layer. The FC layer will combine the extracted features into high-level features to learn valuable non-linear combinations. Finally, the *softmax layer* is employed to transform these high-level features into probabilities for each class. By leveraging properties from the exponential function e^x, the input, which can be any real number, is transformed to a positive real number. In particular, the exponential function transforms differences in the input into their ratios.

$$e^{(x_1-y_1,\ldots,x_n-y_n)} \longrightarrow (\frac{e^{x_1}}{e^{y_1}}, \ldots, \frac{e^{x_n}}{e^{y_n}}) \tag{2.3}$$

To transform the features into probabilities, a normalization is made to sum the distribution to one.

$$\text{Softmax}(\vec{x}) = (\frac{e^{x_1}}{\Sigma_{k=1}^{n} e^{x_k}}, \ldots, \frac{e^{x_n}}{\Sigma_{k=1}^{n} e^{x_k}}) \tag{2.4}$$

To enable learning for a supervised network, a *loss function* is employed. Its objective is to calculate a score based on the deviation in the softmax prediction from the true label. In other words, the loss function is initiating the learning in the network by finding how far off the target the current prediction is. The *Cross-Entropy* loss function, also known as logarithmic loss, is the most commonly used loss function in classification models. For a multi-class classification problem, the cross-entropy function is expressed as

$$\mathbf{J} = -\sum_{i=1}^{N} y_i \cdot \log(\hat{y}_i) \tag{2.5}$$

where N is the number of samples evaluated, y_i is the ground truth vector and $\hat{y}_i$ is the prediction from the softmax layer in the learning model. Important to notice for the cross-entropy loss is that it only penalizes/rewards the prediction on the ground truth class. That is, y_i works as an activation function for the value of the prediction in $\log(\hat{y}_i)$, so the confidence in other classes will not be considered. In other words, cross-entropy can be regarded as the negative log of the estimated probability of the true class. Since the score is logarithmic it will offer a small score for small differences from the target value and enormous scores for large deviations. The output from the loss function is used to optimize the model loss with respect to the weights in the network layers. An *optimizer* is employed to search for the set of weights that minimizes the loss function, i.e the difference between the prediction and the target. Commonly used optimization functions include stochastic gradient descent (SGD) and adaptive movement estimator (ADAM). Both methods choose random data points

from the loss function to calculate a *gradient* and then move in the steepest direction with the aim of minimizing the loss function with respect to the model parameters. Mathematically, this optimization can be expressed as

$$\mathbf{W}^{(k+1)} = \mathbf{W}^{(k)} - \eta \cdot \frac{\partial}{\partial \mathbf{W}^{(k)}} \mathbf{J}(\mathbf{W}) \tag{2.6}$$

Where $\mathbf{J}(\cdot)$ represents the loss function employed, $\mathbf{W}$ is the set of weights in the network and η is the learning rate. Based on the current set of weights and the learning rate η, the optimization function finds the steepest direction to go and how big steps to take to update the weights, $\mathbf{W}^{(k+1)}$, such that the loss function is minimized. A challenge with optimization is non-convex loss functions that can halt the optimization at local minima or saddle points. In order to overcome these challenges, hyperparameters, such as *learning rate* and *batch size*, needs to be fine-tuned. Moreover, optimization functions such as *Adam* and *Adagrad*, do also employ individual and adaptive learning rates for each weight making them converge faster and also more robust against local minima.

2.2.2 Regularization

Regularization, in machine learning, is a set of techniques that aim to optimize the performance of the learning model by promoting generalization and avoiding overfitting.

- **Dropout** is a regularization technique that randomly switches off neurons in a deep learning network at run time. A neuron is switched off when the ability to output a result from the activation function to the next layer in the network is removed. In other words, with different configurations of neurons at each training iteration, a single model can simulate having several different architectures. This is effectively making each neuron more important in the network and removing any large weights caused by dominant neurons. As a result, each neuron becomes less sensitive to input changes which in turn results in a model that is generalizing better.

- **Data augmentation** is a regularization technique to artificially increase the amount of training data provided to a machine learning model. The augmentation is achieved by applying a set of transformation functions to the existing pool of samples so that the machine learning model will learn from new variants of the existing images at each round. Typical methods of data augmentation include random rotation, translation, and addition of jitter. With data augmentation, the concept of static datasets becomes more dynamic and it artificially increases

the number of images for the machine learning model to learn from. However, data augmentation does not increase the number of features in the images, so rather than replacing data gathering, it helps exploit the full potential in the existing dataset. Its performance enhancement is especially prominent on smaller datasets where the risk of overfitting on the training data is a potential issue, hence it improves the generalization ability of machine learning models trained on a small set of features.

- **Early stopping** is an effective, yet simple regularization technique applied in deep learning. By monitoring the training and validation error described in section 2.1, it aims at stopping the training process when the learning model starts to overfit on the training data.

2.2.3 Transfer learning

As will be further elaborated in section 2.4, many applications of machine learning suffer from constrained amounts of data to learn from. Moreover, image classifiers and other machine learning models will often degrade in performance when employed in a different domain from what it was trained in. In machine learning terminology, the domain in which a model is trained is referred to as the source domain, whereas the domain it is deployed in is referred to as the target domain [56]. To overcome the challenge of limited data in the target domain, a model will be trained on samples from a related source domain. Then, the weights and parameters of this model are incorporated into a new model which is employed in the target domain. The idea is to use the obtained knowledge from the source domain in the early layers of the new neural network to identify high-level features from the target domain. Further, training the subsequent layers of the model on data from the target domain optimizes it for predictions in this new domain. This concept of sharing knowledge, i.e weights and parameters, between learning models is known as *transfer learning*. In effect, this concept reduces the demand for training data and elevates the initial knowledge of the learning model.

2.2.4 Evaluation measures

Quantitative measures are needed to evaluate the performance of a classifier. It is in general not possible to measure the overall performance of a classifier since various metrics are weighted differently depending on the application. That is, for some applications, such as medical imaging, it is more acceptable to have false positives (FP) than false negatives (FN), hence recall (2.8) is the best measure. However, for the

application of plankton classification, it is crucial to measure the number of images being classified correctly, hence accuracy or balanced accuracy is mainly employed as the evaluation metric. Furthermore, the properties of the training set presented to the model will also lead to different results depending on the evaluation metric. The *accuracy paradox* is an example of a situation where a metric leads to a bias in the accuracy evaluation. When presented with an unbalanced dataset, the model will predict the most prominent class correctly multiple times without necessarily learning the underlying difference from the other classes. Thus, for unbalanced datasets, one should consider balanced accuracy (2.10), over accuracy (2.7) as it will give a lower score if the model is unable to predict sparse classes.

- **Accuracy**

$$\text{Accuracy} = \frac{TP + TN}{Total} \tag{2.7}$$

  A measure of how many classes that got the right prediction among all predicted.

- **Recall**

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.8}$$

  A measure of the proportion of samples that is correctly classified as the target class. Also known as true positive rate (TPR).

- **Specificity**

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{2.9}$$

  A measure of the proportion of samples not belonging to the target class that is not predicted as the target class either. Also known as true negative rate (TNR).

- **Balanced accuracy**

$$\text{Balanced accuracy} = \frac{TPR + TNR}{2} \tag{2.10}$$

  An average of recall 2.8 and specificity 2.9. Handles imbalanced datasets better than accuracy 2.7.

2.3 Data labeling problem

The success in the field of computer vision is largely related to the development of CNNs as described in section 2.2.1. However, the success comes at the price of immense

amounts of labeled data needed for training the networks due to increasingly deep networks with large amounts of parameters [27]. It is non-trivial to quantify the number of labeled images needed for each class since it will vary with the task complexity, the learning model, and other parameters. That is, *data augmentation* and *transfer learning* could both reduce the amount of data required. However, how aggressively the data can be augmented is dependent on the dataset, and transfer learning would require some similarities between the learning tasks to be advantageous. Further, the number of features represented in the data pool are often considered more important than the exact number of data samples. However, the former quantity tends to increase with the number of data samples, hence more data are generally considered to be better.

Furthermore, labels categorizing cars and pedestrians can be acquired at a fairly low cost. For more complex domains such as radiology or biology, a domain expert is often needed to label samples. This drastically increases the cost of acquiring labels and also puts a burden on the employed domain experts. Nevertheless, as discussed in [14], the resulting manual classification is imperfect and prone to errors and multiple domain experts are preferred to achieve high classification accuracy and confidence in the labeled dataset. With the increased adoption of machine learning for image classification, effective methods for the construction of labeled datasets are essential to fully leverage novel models and algorithms. Active Learning (AL) is a technique that has been proposed to address this challenge by only labeling the most informative samples.

2.4 Active Learning

Active learning is a type of semi-supervised learning that aims at mitigating the burden of manual labeling on domain experts and speed up the construction of labeled datasets. By exploiting a non-uniform information distribution among images in a data pool [61], active learning aims to find the most informative samples and query them for manual labeling. By constructing a dataset leveraging samples with large amounts of information, [12] showed that an employed classifier could achieve equal classification performance as if it was trained on the full dataset. Active learning has previously been applied to natural language processing (NLP) [54] and image segmentation [64] in addition to other areas. However, since those areas of application are outside of this scope, the following presentation of active learning modes and methods will mainly consider the application for image classification. Moreover, this section will start with a presentation of general concepts and modes in active learning before presenting active learning for deep learning, namely deep active learning (DAL).

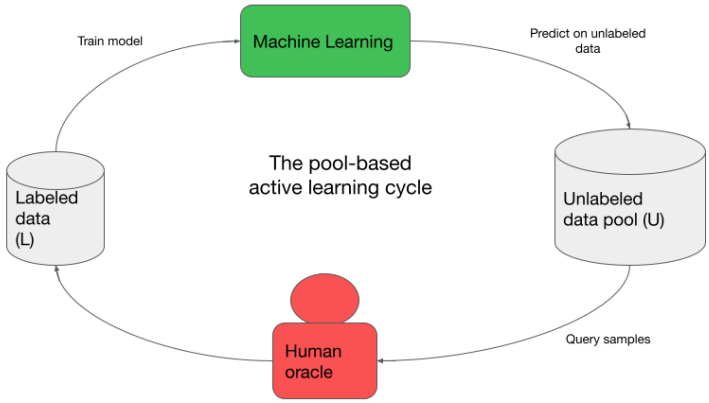


Figure 2.7: The pool-based active learning cycle.

2.4.1 Active learning cycle

The aim of minimizing human effort in data labeling has been around for many years, hence there exists a broad literature on active learning, including research conducted before the widespread adoption of CNNs in the field of computer vision. The reader can refer to a survey presented in [53] for a summary of the early work conducted in the field of AL. The data flow in an active learning approach, as illustrated 2.7, is common among most of the approaches in the literature and works as a backbone in active learning. Commonly, the active learning cycle is initiated with a small pool of labeled data used to warm up an employed ML model. After an initial training round, the model is then employed as a classifier on an unlabeled data pool, and the results from this classification are often used as the foundation for a query of new data points. A pre-defined number of data points is then queried to a human oracle for labeling and added to the pool of labeled data. This AL cycle repeats until a pre-defined labeling budget is exhausted, or an early stopping criterion is met. The latter is applied if the manual labeling effort stops giving significant performance enhancement for the machine learning model.

2.4.2 Sampling modes

Existing AL models in the literature can be classified based on the unlabeled data readiness, the number of points queried and the strategy employed for querying. In some cases, the data gathering is performed once, whereas in other cases the data arrives as a stream of data points. In other words, when the data arrives in streams the AL model is considered as a *stream-based model* [35], whereas a *pool-based model* otherwise [38]. An application of the former is the construction of a machine learning training set based on incoming radiology images of a newly discovered disease, as proposed in [63]. When time and expert capacity is limiting factors in the construction of the training set, images must be queried continuously for a human oracle, i.e radiologist, based on the amount of information they provide to the image classifier. On the other hand, for pool-based active learning, samples are captured in large batches, e.g with a video camera as described in [45], and the overall goal is to create a dataset while minimizing the human effort and time consumption. An illustration of the data flow for the pool-based active learning is illustrated in figure 2.7. An important distinction between the aforementioned modes is that the latter queries samples based on an evaluation of all samples in an unlabeled data pool whereas the former needs to make an independent judgment for each arrived data point. The application of the work presented in this thesis is only concerned with pool-based active learning with batch sampling for image classification. Hence, future references to active learning will imply this mode unless other is specified. Pool-based active learning and active learning will be used interchangeably hereupon. Further, the AL models' mode of sampling varies between batch-mode [7] or single-mode [38] depending on the number of data samples queried in each cycle. With the recent development of CNNs, as described in 2.2.1, batch-mode sampling has become increasingly relevant as it is not computationally feasible to update a large network with single data points nor are single data points likely to give a significant update to the model parameters, causing series of correlated queries. As will be elaborated in the upcoming sections, the most important distinction between the above-mentioned sampling modes is their prioritization between informative and representative samples.

2.4.3 Machine learning knowledge quadrant

For a machine learning model, one can divide the models' knowledge into four different categories based on their current knowledge and awareness of the available training data. This grouping of knowledge, illustrated in table 2.1, is motivating the query strategies for active learning frameworks. As described in section 2.4.1, methods of active learning are often initiated with a small pool of labeled images, i.e samples,

which will contribute to the current model state. A machine learning model trained with this pool is parsed through unlabeled samples to find the most relevant *unknowns* for the model to learn from.

	Knowns	**Unknows**
Known	Current model state	Non-confident predictions from the model
Unknown	Transfer learning	Gap in model knowledge

Table 2.1: Knowledge quadrant for machine learning.

Since active learning aims to minimize the amount of effort needed for data labeling, it is important to address the different quadrants in figure 2.1, to utilize the data distribution best possibly. By employing representative sampling, the learning model can mitigate any gaps in knowledge and from informative sampling non-confident predictions can be overcome. Lastly, transfer learning can be applied to give the learning model a head start by incorporating initial knowledge, i.e pre-trained weights, into the model. In the deep active learning section 2.5, the individual methods will be further elaborated for use in a deep learning setting.

2.5 Deep active learning

The development of CNNs has brought high classification accuracy, however at the price of increased amounts of manually labeled data needed for training. The promise of removing the bottleneck of this manual labeling in the construction of these datasets has brought a surge in DAL research. However, with the introduction of CNNs in the field of computer vision, traditional methods of active learning have shown to struggle and often perform worse than random benchmark sampling (RBS). In particular, finding samples that the machine learning model finds informative has become more challenging due to the more complex structure of neural networks. However, there exists a broad literature on deep active learning approaches, and in general, they can be split into informative, representative, and hybrid approaches. These, in addition to some less focused approaches, will be further elaborated in the following sections.

2.5.1 Informative approaches

The informative mode of active learning aims to find the samples in which the *image classifier* finds most informative. In other words, features the model knows it is uncertain about, corresponding to the second quadrant in table 2.1. Important to notice is

that the samples in which the model finds the most informative, not necessarily are the overall most informative samples. This leads to the challenge of *transferability* which will be discussed in later sections. There exist a broad literature on informative-based active learning, and several heuristics for finding samples in which the learning model finds informative have been proposed, the coming sections will provide a description of the main categories.

Distance based methods

Distance-based methods aim to find samples that lie at the proximity of the inter-class decision boundary. Samples that lie in this border area are considered to be informative for the machine learning model as they can provide information to fine-tune the classification decision boundary, as illustrated in figure 2.9. [57] proposed a distance-based method for an SVMs classifier. However, as it is feasible for an SVM, it is a more complex operation for dense CNNs. Nevertheless, to transfer this approach to CNNs, [16] proposed a way of measuring the distance by making *adversarial attacks*, that is perturbing the pixel values in the input image until the employed image classifier changes the classification. By ranking the amount of perturbation needed for a change in classification, one can obtain a proxy for how far a given sample is from the decision boundary. The adversarial attack approach is based on the DeepFool algorithm proposed by [42]. The idea is that the orthogonal projection of a sample X_i onto the hyperparameter plane representing the inter-class decision boundary is corresponding to the minimal perturbation needed to change the decision of the classifier. This orthogonal projection can be calculated as

$$\frac{f(x_i)}{\|\nabla f(x_i)\|} \cdot \nabla f(x_i) \tag{2.11}$$

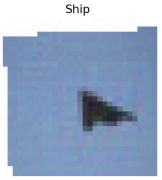
where $f(x_i)$ is the output from the softmax layer of the CNN and $\nabla f(x_i)$ is the calculated gradient from the loss function. This calculated projection is added to the sample as a perturbation before the image is re-classified.

(a) Original image and its model prediction.

(b) Perturbed image and its new model prediction.

(c) Total perturbation.

(d) Original image and its model prediction.



(e) Perturbed image and its new model prediction.

(f) Total perturbation scaled up with a factor of 10 for the purpose of visualization.

Figure 2.8: Two samples showing how the images are perturbed to push them over the decision boundary. The first row presents a sample from the first round of active learning, whereas the second row presents a sample from the last round of active learning.

One can observe from two samples drawn from the DeepFool method in an active learning cycle how the decision boundary has changed from the initial to the final active learning cycle. For the image in figure 2.8a, the classifier has not seen enough features to be confident about its classification and is conducting two wrong classifications. For the image in figure 2.8d however, the classifier is accurately classifying it, but based on the small perturbation needed to change its classification, as illustrated in figure 2.8f, the classifier was not confident in its prediction. This uncertainty indicates a sample at the proximity of the decision boundary, as illustrated in figure 2.9. The algorithm for the DeepFool procedure described is presented in algorithm 1. While the process in the algorithm will continue for several iterations, in particular for each data point x_i in the unlabeled data pool D^U, the explanation hereafter will be restricted to a single data point for simplicity. To begin with, the sample x_i is classified by the employed classifier, as described in line 3. Thereafter, projections to other hyperplanes corresponding to the other classes are approximated in lines 6-8. For the smallest distance found, a projection to the corresponding hyperplane is calculated in line 10. This projection is then added to the sample x_i as a perturbation, and a new classification is conducted in

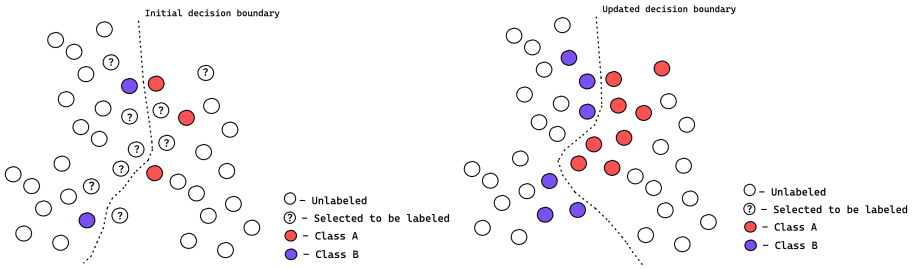


Figure 2.9: Informative sampling at the proximity of the inter-class decision boundary helps fine-tune the decision boundary.

lines 13 and 14, respectively. This process continues until the data point x_i changes its classification, as illustrated in figure 2.8, or until a maximum number of iterations has been reached. All perturbations are then sorted in a list D. In the approach by [16], a pre-defined number of the least perturbed samples were queried for labeling. For the proposed framework in this thesis, a sub-modular heuristic will be applied instead to avoid correlated queries, this will be further elaborated in section 5.1.

Ensemble methods

Ensemble methods of informative learning aim at comparing the opinion of multiple network architectures to find samples on which they disagree. Often the disagreement criteria employed is either entropy, indicating multiple different classifications, or margin sampling where the classifiers are usually split between two prominent classes. In the literature, there are proposed two main ways of doing ensemble-based informative sampling. In [8], multiple different network architectures are trained in parallel on the same training set, before conducting separate predictions on new, unseen data. While this achieves a broad range of perspectives on the same data, it is computationally demanding to train multiple different networks, especially with increasing amounts of training data and model parameters. A conceptually similar approach however was proposed by [18], their method, adopting the work in [29] to deep learning, rely on *Monte Carlo (MC) dropout* of neurons to effectively sample multiple different network architectures. This brings a two-fold measure of uncertainty. First, by making multiple runs and registering the different networks' prediction on the input X, some of the network configurations associated with a set of neurons will be confident in the wrong category. Thus, by labeling the images they are confidently wrong about, the associ-

Algorithm 1 DeepFool: Multi-class adversarial attack.

Require: Unlabeled samples D^U
Require: Learning network hyper-parameters $\mathcal{H}$
Require: Empty list of distances **D**
Require: Number of classes N
$V_l = \inf$
for $x_i \in D^U$ **do**
  $ny = py = \textsc{Predict}(x_i)$
  while $py = ny$ **do**
    for $x_j \in x_{\{1,..,N\}}$ **do**
      $W_i \longleftarrow \nabla x_i - \nabla x_j$
      $F_i \longleftarrow f(x_i) - f(x_j)$
      $V_i \longleftarrow \frac{F_i}{\|W_i\|} \cdot W_i$
      if $V_i \leq V_l$ **then**
        $R_i \longleftarrow \frac{V_i}{W_i} \cdot W_i$
      end if
    end for
    $\eta \longleftarrow \eta + r_i$
    $py = \textsc{Predict}(X + \eta)$
  end while
  $\mathbf{D} \cup r_i$
end for
return **D**

ated neurons can be corrected. Moreover, for other inputs, the ensemble of network samples may in general be uncertain on the label, causing a high entropy in the output, hence requiring the sample to be labeled. While the aforementioned method [18] was restricted to single queries, it was expanded to batch mode sampling in [34]. However, research conducted by [9] comparing the aforementioned approaches reported that ensemble-based approaches, as proposed in [8], outperforms other methods of uncertainty estimation and in particular MC dropout. Their experiments compared an ensemble of five networks with an MC dropout model with 25 forward passes. A method combining ensemble methods with MC dropout was proposed by [49]. The paper addressed challenges related to mode collapse causing overconfident predictions in methods similar to [18].

Softmax based methods

A large number of the informative active learning approaches proposed in the literature have been based on the softmax layer of a neural network as a proxy for the model uncertainty. In general, three different heuristics of applying the probability scores from the softmax layer have been studied in the literature.

- The **least confidence** strategy aims to find the samples which is predicted with the lowest confidence. The expression $p(y_i = j|x_i; \theta)$ describes the probability for variable x_i to belong to the j$^{\text{th}}$ category. Samples with low score indicates low certainty from the model.

$$\text{LC}_i = \max_j P(y_i = \text{j}|x_i; \theta) \tag{2.12}$$

- The **margin sampling** approach aims to find the samples where the margin between the two most probable classes is the smallest. This margin is found by subtracting the second highest class probability $P(y_i = j_2|x_i; \theta)$ from the highest class probability $P(y_i = j_1|x_i; \theta)$.

$$\text{MS}_i = P(y_i = j_1|x_i; \theta) - P(y_i = j_2|x_i; \theta) \tag{2.13}$$

- The **entropy sampling** strategy finds the samples where it is the most disorder in the predictions. That is samples where no category is prominent in the probability distribution. Higher values of entropy mean more disorder and consequently higher uncertainty.

$$\text{EN}_i = -\sum_{j=1}^{m} P(y_i = j|x_i; \theta) \log P(y_i = j|x_i; \theta) \tag{2.14}$$

In a paper by [62], the authors proposed a method leveraging the softmax layer of the model to find samples in which the model is uncertain. Their proposed method also employed a cost-effective module to find high-confidence samples for pseudo-labeling. These samples, with confidence above some threshold, were added to the training pool for one iteration to increase the robustness and accuracy of the model, however at risk of being erroneously labeled. The results in the paper suggest that the most effective sampling heuristic was a combined approach where each of the aforementioned methods selected a certain number of samples for the query. Additionally, the reported results did also show a significant performance enhancement in terms of classification accuracy when employing the cost-effective module. Nonetheless, research has shown that these softmax probabilities often work as a bad proxy for the confidence of neural networks [23, 50], and will often lead to worse performance than RBS. This can partially be explained by overconfidence in the predictions due to the applied exponential function in the softmax layer described in section 2.2.1. In particular, the exponential function has the property of turning addition into multiplication, that is $e^{a+b} = e^{a \cdot b}$. It is trivial to show that this property makes the softmax layer translation invariant, hence not a reliable measure for the uncertainty of the neural network.

$$\begin{aligned}
\text{Softmax}(x_1 + a, ..., x_n + a) &= \left(\frac{e^{x_1+a}}{\sum_{k=1}^{n} e^{x_k+a}}, ..., \frac{e^{x_n+a}}{\sum_{k=1}^{n} e^{x_k+a}}\right) \\
&= \left(\frac{e^{x_1} e^{a}}{\sum_{k=1}^{n} e^{x_k} e^{a}}, ..., \frac{e^{x_n} e^{a}}{\sum_{k=1}^{n} e^{x_k} e^{a}}\right) \\
&= \left(\frac{e^{x_1}}{\sum_{k=1}^{n} e^{x_k}}, ..., \frac{e^{x_n}}{\sum_{k=1}^{n} e^{x_k}}\right) \\
&= \text{Softmax}(x_1, ..., x_n)
\end{aligned}$$

Redundant sampling

A challenge with pure informative sampling in batch mode AL is the labeling of redundant samples and a lacking utilization of the full data distribution. This challenge is a result of the sampling process not being batch-aware, i.e there is no knowledge transfer among the queried samples in which areas are being covered, and informative sampling often tends to query multiple samples from the same area of uncertainty. This can be observed in figure 2.10 where 200 samples have been queried with an adversarial attack AL strategy, as described in 2.5.1, from a pool of images from the CIFAR-10 dataset. It can be observed that the AL strategy is querying a lot of images from the same areas of the feature space, and consequently lack covering in other areas. The most significant information for the classifier will be provided by the first images queried

from new areas whereas the later queries will often tend to give redundant information. To minimize the manual labeling effort, it is desirable to avoid this redundant querying. By lowering the number of queries for each round, the number of correlated samples would resultingly be lowered, however, at the expense of a higher computational effort, this challenge will be further discussed in section 7.2.

Figure 2.10: 200 samples queried with the DFAL [16] informative approach. The different colored data points represent the images of the ten different classes from the CIFAR-10 dataset. With the T-SNE algorithm [58], the images are projected onto the two-dimensional feature space.

2.5.2 Representative approaches

The representative mode of active learning aims to exploit the latent space of the available unlabeled samples to best capture the data distribution, as illustrated in figure 2.11. The information in the queried representative samples is often related to a gap in the model knowledge, represented by the fourth quadrant in table 2.1. A large number of methods for finding such representative samples have been researched in the literature and will be described in the coming sections.

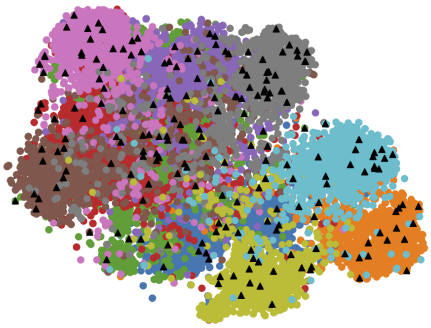


Figure 2.11: 200 samples queried with the core-set [52] representative approach. The different colored data points represent the images of the ten different classes from the CIFAR-10 dataset. With the T-SNE algorithm [58], the images are projected onto the two-dimensional feature space.

Core-set approach

The *core-set approach* aims at selecting data points such that a model trained over the queried data points is competitive with a model trained over the full dataset. By regarding the query process of active learning as a core-set selection problem, the strategy of querying data points can be formulated as an optimization problem where the target is to select data points that cover the full feature space to minimize the *core-set loss*. As described in section 2.1 an ML model aims to minimize the training and classification error. However, in the core-set selection, and AL in general, another error term, i.e core-set loss, is introduced. The core-set loss represents the information that is not captured with the selected data points, in other words, the difference in the data distribution between the full dataset and the queried samples. Formally, given a set of data points defined over a feature space X, a corresponding set of labels $Y = \{1, .., C\}$ and a loss function $\mathcal{J}(X, Y, A_{s^0 \cup s^1})$ where $A_{s^0 \cup s^1}$ is the parameters of the machine learning model, the optimization problem of a core-set approach can be expressed as

$$\min_{s^1 : s^1 \leq b} \underbrace{\overbrace{\frac{1}{n} \sum_{i \in [n]} \mathcal{J}(\mathbf{x}_i, y_i; \mathbf{A}_{s^0 \cup s^1})}^{\text{Loss over all data}} - \overbrace{\frac{1}{\mathbf{s^0} \cup \mathbf{s^1}} \sum_{j \in \mathbf{s^0} \cup \mathbf{s^1}} \mathcal{J}(x_j, y_j; \mathbf{A}_{\mathbf{s^0} \cup \mathbf{s^1}})}^{\text{Loss over selected samples}}}_{\text{Core-Set Loss}} \tag{2.15}$$

Where n is the size of the full dataset, b is the labeling budget, s^0 is the initially labeled samples, and s^1 is the queried samples. The optimization problem aims to query b

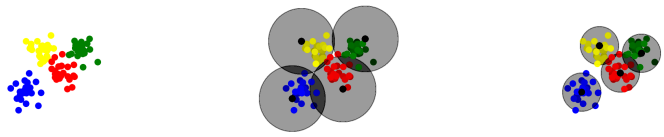
samples to minimize the overall core-set loss. In [52], the authors proved that this optimization problem could be upper bounded by a constant δ_s representing the largest distance from any single data point to its nearest cluster, i.e the radius of the largest cluster. Consequently, minimizing the core-set loss is equivalent to optimizing the K-center problem expressed as

$$\min_{s^1:s^1 \leq b} \max_{i} \min_{j \in s^1 \cup s^0} \triangle(x_i, x_j) \tag{2.16}$$

Where $\Delta(x_i, x_j)$ represents the Euclidean distance between the data points x_i and x_j. As this problem is NP-hard, a sub-optimal solution is found by a greedy algorithmic approach as described in algorithm 2. This greedy method is proven to have a solution such that

$$\max_{i} \min_{j \in s^1 \cup s^0} \triangle(x_i, x_j) \leq 2\, X\, OPT \tag{2.17}$$

is satisfied, where OPT is the optimal solution to the optimization problem in 2.16 [28]. The authors in [52] optimized this solution by applying a mixed-integer programming (MIP) subroutine. However, while increasing the computational effort in finding a solution, their reported performance enhancement was small. A further description of the optimization is outside the scope of this thesis, however, the reader can refer to the paper by [52] for further elaboration on the formulation of the optimization problem. Nonetheless, two experiments comparing the core-set approach with and without the optimization module are presented in section 6.1.



(a) 4 different classes in 2D feature space.

(b) Greedy solution to find cluster centers.

(c) Optimal solution on cluster centers.

Figure 2.12: Three steps of the core-set approach. The radius of the ball in (b) and (c) represents the upper bound on the core-set loss. That is, the error of not labeling all samples in the dataset.

The concept of choosing cluster centers to minimize the core-set loss is illustrated in figure 2.12. The figure presents four different classes projected to the two-dimensional

space. In figure 2.12b a greedy solution with a resultingly large δ_s is shown, whereas in figure 2.12c an optimal solution with a significantly smaller δ_s is presented. For the former, the upper bound on the core-set loss is by visual inspection twice the optimized solution. Further, it is trivial to observe that as the number of clusters approaches the number of samples n, the core-set loss (2.15) becomes zero.

Algorithm 2 MinMax: Greedy geometric approach.

Require: Unlabeled data X_i
Require: Initial labeled data S^0
Require: Labeling budget B
  $S = S^0$
  while $|S - S^0| \leq B$ **do**
    $u = \arg\max_{i \in [n] \setminus S} \min_{j \in S} \Delta(x_i, x_j)$
    $S = S \cup \{u\}$
  end while
  return $S \setminus S^0$

The algorithm for solving equation 2.16 with a greedy approach is described in algorithm 2. A pitfall with the core-set approach is the reliance on extracted features from the unlabeled data pool. To solve the optimization problem described in equation 2.16, the dimensions of the images needs to be lowered to make the computation feasible. To successfully query representative data points, it is important that this feature extraction best represents the underlying data distribution of the selected images. Another challenge with the core-set approach is the propensity to query from sparse areas which often can represent outliers in the unlabeled data. To overcome this, the sampling strategy should be aware of sparse, outlier regions which can confuse the learning model.

K-means approach

K-means is another clustering technique that has been proposed as a metric for representative active learning approaches. It aims at minimizing the intra-class variance of the cluster by minimizing the average squared distance to an approximated cluster center for the data points within the same cluster. Different from the K-center approach previously described, this is achieved by calculating a cluster center which represents the average of all the points in the cluster. Formally, this minimization can be expressed

as

$$\sum_{i=0}^{n} \min_{\mu_j \in C} (\|x_i - \mu_j\|^2) \tag{2.18}$$

where n is the number of data points and μ_j is the mean of all samples in the cluster C, i.e the cluster center. Further, to give the algorithm a better initialization, [6] proposed *k-means++*, a technique that instead of choosing k randomly chosen cluster centers, randomly picks one initial center and then strategically chooses $k - 1$ more centers. Each new initial cluster center is chosen with a probability of $D(x)^2$, where $D(x)$ is the distance for any given point x to its nearest cluster. Hence, the k initial cluster center gets spread out and results have proven this strategic initialization of the k-means algorithm to give more optimal clusters and faster convergence speed [6].

Bayesian sparse-set approach

In a paper proposed by [48], the authors aim at querying a batch of data points $\mathcal{D}'$ at each AL cycle such that the data log posterior of the initially labeled samples $\mathcal{D}_0$ combined with $\mathcal{D}'$ best possible approximates the data log posterior of the full data distribution $\log p(\theta|\mathcal{D}_0 \cup \mathcal{D}_p)$ where $\mathcal{D}_0$ and $\mathcal{D}_p$ is the initially labeled samples and the pool of unlabeled samples, respectively. However, as calculating the full data posterior is not possible for the unlabeled dataset, the authors employ the expectation of the predictive posterior distribution. Resultingly, their query function is based on choosing samples that minimize the difference between this expected full posterior and the resulting posterior from the queried data points. Their method is similar to the previously described core-set approach in that they are trying to approximate the complete data distribution with a subset of the samples. However, an important distinction is that while the core-set approach achieved this by solving the k-center problem described in algorithm 2, the proposed method is trying to resemble the complete data posterior. Hence, due to the geometric approach, the former needs a learned representation of the unlabeled data which is not needed for the probabilistic Bayesian sparse set approach.

2.5.3 Hybrid approaches

Hybrid modes of active learning aim at combining the metric for informative sampling with the metric for representative sampling to both utilize the full data distribution and identify samples the classifier finds informative. Hence, hybrid mode active learning addresses the full column of unknowns in figure 2.1. This hybridization aims to overcome many of the challenges that each of the metrics in separate suffers from,

among them is avoiding queries from areas where the model is uncertain. Moreover, by not exclusively relying on the model uncertainty, the *transferability*, how well it performs with different learning models and domains, is increased. In a hybrid method proposed by [7], the authors employ the induced gradient of the predicted category, described in 2.2.1, as an uncertainty measure for the model. The rationale behind this choice of uncertainty metric is that the magnitude of the gradient increases with the deviation between the prediction and the target value. Since this gradient is a vector that is also employed to train the network through backpropagation, it is possible to use the direction of the vector to incorporate diversity in the queried samples. In the proposed framework, the authors employ the *k-means++* strategy to select gradient vectors that best enhances the diversity in the queries. Another way of combining a representative and informative metric was proposed by [30]. Their method, *Active Learning by Learning*, is a hybrid approach, inspired by the multi-armed bandit problem [59], that chooses between different sampling strategies at run time. The method will for each AL cycle explore the performance of different sampling methods and exploit the one with the best performance. Furthermore, hybrid methods that directly leverage the probability distribution of the softmax layer have also been proposed. In [67], the author proposed a hybrid method with a weighted trade-off between the informative and representative methods. To incorporate uncertainty, each sample is given a weight based on its score from the softmax layer margin sampling, described in equation 2.13. This score is combined with an additional representative score from *k-means* sampling. The paper suggested that "diversity-enhancing approaches slightly or significantly outperform the strong baseline of uncertainty sampling" [67].

2.5.4 Other approaches

Other AL approaches not as focused as the above-mentioned have also been proposed in the literature. One example is the **task-agnostic approaches** which is designed to be independent of the employed learning model and domain it is applied in, hence increase the *transferability*. Such an approach, employing an auxiliary classifier to predict the *learning-loss*, was proposed by [65]. The proposed method use extracted features from the image classifier to train a separate learning model to predict the loss from unlabeled samples. The key idea is that a high loss prediction indicates samples that the classifier is uncertain about. The separate model is trained on an initial labeled pool comparing the predicted loss with the actual loss calculated by the employed image classifier. This module is adaptable to any learning model as it only employs the extracted features from the intermediate layers of the model. A similar approach of training a separate network for the task of querying informative samples was proposed by [21]. Their proposed *discriminative active learning* method, is based on training

a binary classifier to select samples such that the difference between the unlabeled and labeled dataset becomes indistinguishable. This is achieved by querying samples that brings features not seen in the labeled dataset, hence informative to the image classifier. Another conceptually equal approach was proposed in [55]. Similar to the above-mentioned approach, they aim at training an adversarial network to discriminate between labeled and unlabeled samples. Their proposed method employs a variational autoencoder (VAE) to learn a latent space for the adversarial classifier to predict from. An important distinction from the method proposed in [21] is that the VAE is trained to fool the adversarial classifier to believe that all samples are from the labeled data, whereas the adversarial network is trained to discriminate between dissimilarities in the feature space. These methods of constructing labeled datasets to approximate the unlabeled data distribution is conceptually similar to the aforementioned methods in 2.5.2 and 2.5.2.

By addressing the third quadrant in table 2.1, **transfer learning** adds initial knowledge to the target model without increasing the labeling effort, as described in 2.2.2. In a method proposed by [20], transfer learning is applied in the active learning approach. Their proposed method will gain satisfying accuracy on a pre-trained network and then apply active learning techniques to gain additional accuracy over the long tail, that is apply active learning. to fine-tune the model. Their proposed method employed farthest-first traversal, which is conceptually similar to the aforementioned K-center algorithm, as an active learning strategy. Another method related to the limited availability of images in a target domain was proposed by [19]. The proposed method will initially train a model on images from a source domain before fine-tuning the model on images from a target domain. Secondly, a domain expert is then employed to label queried images that represent uncertain and abundant patterns from the target domain.

Chapter 3

Related work

The related work chapter presents research and work related to this thesis. To begin with, previous research related to minimizing manual effort for plankton taxa labeling is presented. Then, similarities between the proposed framework and approaches from the literature are presented.

3.1 Approaches to minimize manual effort for plankton taxa labeling

Classification of planktonic species with machine learning has been widely studied in the literature with promising results [39, 15, 68, 66]. However, these methods are developed based on a readily available data pool of labeled samples. The recent development of data-intensive learning networks has created a surge in research due to the rate of data gathering surpassing the human annotation rate. Methods proposed in the literature cover a broad range of strategies from annotation-free learning to strategic labeling with active learning.

3.1.1 Annotation-free learning of plankton for taxa classification

In the paper [45], the authors proposed a framework for annotation-free learning of plankton for classification. Their proposed method consists of an unsupervised and a supervised part, as described in 2.1. Firstly, by employing a partition-energy-based technique they were able to approximate the number of classes the dataset should

be partitioned into. Following this procedure, three different unsupervised clustering algorithms were employed to cluster the data samples. In their results, a fuzzy variant, i.e each data point can belong to more than one cluster, of the previously described k-means clustering algorithm proved to give the best result. To annotate the plankton samples, two different methods were applied. Having labeled the different clusters, the first method leverage these labels to label new samples based on their nearest cluster. The second method, which also proved to give the best result in terms of classification accuracy, applied the labels obtained from the unsupervised partitioning algorithm to train a supervised classifier. Further, to account for novel and unseen samples the authors proposed an anomaly detector based on an SVM binary classifier. Anomalies were defined to be samples that deviated, above a given threshold, from the average learned features for a particular class. Having one anomaly detector trained per class in the framework, the authors successfully identified new and unseen plankton categories in the unlabeled dataset.

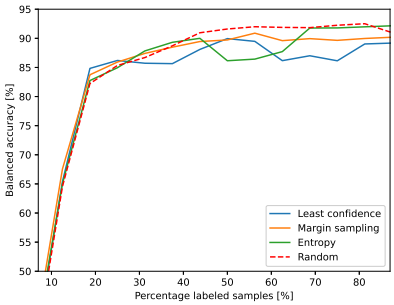
3.1.2 Efficient clustering-based plankton annotation

In [51], the authors proposed a method for efficient plankton annotation by embedding unsupervised clustering in the annotation process. The paper addresses the challenge for a classifier to represent all possible planktonic classes present in an unlabeled dataset. The authors' main hypothesis is that a deep convolutional neural network is sufficiently able to learn distinctive features in order to partition plankton images based on their features. The proposed approach consists of multiple steps. Firstly, relevant features are extracted with a trained feature extractor. Secondly, the data points are clustered and visually pure clusters are accepted for further growing while impure clusters are not. A cluster is considered visually pure if the dominant amount of images in the cluster belongs to one class. For each iteration, the minimum cluster size, i.e number of images in a cluster, is lowered in order to allow the model to create smaller and purer clusters. The last step in the proposed approach includes agglomerative clustering to create a hierarchical structure of the planktonic data. Further by manual inspection, similar clusters are merged together and unique clusters are labeled according to the desire of the user. Their reported results claimed to have achieved a classification precision close to 0.9 for 1.08M images out of a pool of 1.2M.

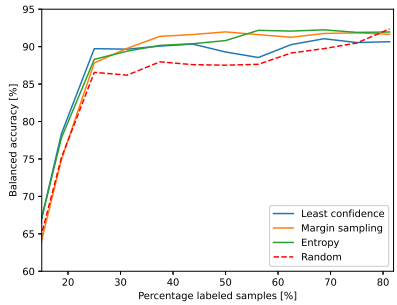
3.1.3 Active learning on the planktonic domain

Another field of related work is plankton-specific active learning. In [40], the authors proposed an AL method using multi-class support vector machines (SVM). In the proposed method, least confidence sampling 2.12 and margin sampling 2.13 based on

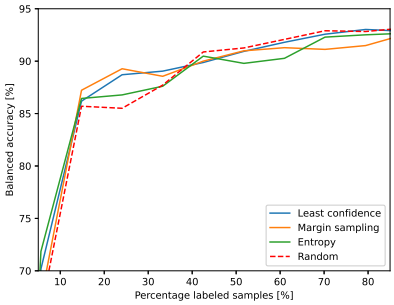
the SVMs decision function were employed to decide which samples to query. Their method was conducted on 8440 grayscale images of zooplankton divided uniformly into five classes. Their paper reported good results for the proposed method in both single and batch mode settings. Following the development of CNNs, [10] proposed a deep active learning approach using the probability distribution from the softmax layer with least confidence, smallest margin, and entropy sampling as uncertainty metrics. Further, aligned with the work in [62], the proposed method applied pseudo-labeling of high-confidence samples to increase the robustness of the learning model, however at the risk of training on erroneously labeled samples. In their proposed method, AlexNet was employed as an image classifier on two datasets from two different biological environments. They randomly selected 10K and 5K images from the two datasets respectively to analyze the performance of the classifiers. Their results reported no significant difference in classification performance for the uncertainty metrics employed. However, the trials including pseudo-labeled high-confidence samples achieved the best performance by reaching an accuracy of the full dataset with only a third of the samples labeled. Moreover, the above-mentioned AL frameworks were implemented on two different planktonic datasets in a report conducted by the undersigned [25]. The three different query strategies based on the softmax layer; least confidence, margin sampling, and entropy were applied in the experiments. The reported results were aligned with the findings of [10], indicating that the cost-effective module is effectively increasing the classification accuracy in a low-data setting. Further results showed that the RBS performed equal to the active learning strategies when the labeled dataset was small, suggesting that the learning model lacked enough feature knowledge to confidently choose the samples it found most informative. This observation motivated the initial incorporation of the full feature space as done in the proposed framework in this thesis. The results from the report are presented in figure 3.1.



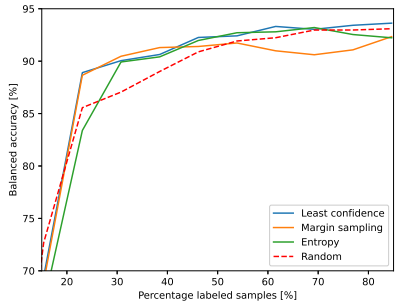
(a) AL on the AILARON dataset.



(b) Cost-effective AL on the AILARON dataset.



(c) AL on the Kaggle dataset.



(d) Cost-effective AL on the Kaggle dataset.

Figure 3.1: Results of applying AL and CEAL to the planktonic domain. From the work conducted in [25].

3.2 Related active learning approaches

A variety of other active learning methods from the literature are related to the proposed framework in this thesis. The core-set approach proposed by [52] is related to the representative metric applied. However, a distinction is that the proposed optimization module is not applied in the proposed framework because of an increased computational effort, however with little gain in accuracy as can be observed in the experimental results in section 6.1. Another pertinent strategy, related to the informative metric, is the DFAL approach proposed by [16]. A difference from their work is the applied

sub-modular heuristic, i.e representative metric, to prevent redundant sampling in the queries. Furthermore, similar hybrid methods have also been proposed in the literature. In [32], the authors combined the margin sampling described in equation 2.13 with the k-means clustering algorithm, described in section 2.5.2, through a trade-off function giving weight to the samples based on both the uncertainty and distinctiveness of the sample. While this is conceptually similar to the proposed approach in this thesis, a significant difference is the use of adversarial attack and data augmentation to improve the informative sampling. Furthermore, by only passing on a few of the samples from the informative method to the representative method, the computational effort of the approach proposed in this thesis is lowered in comparison. Similar hybrid methods employing the softmax layer for uncertainty measure have also been proposed by [33] and [67]. These methods, in addition to [32], are represented by the 'softmax hybrid' strategy in the experiments in chapter 6.

Chapter 4

Datasets

The motivation for the experiments conducted in this thesis was twofold and for this reason, two different grounds for testing were applied in the experiments. Firstly, to compare different methods of active learning, including the proposed CIRAL framework, and measure their performance in terms of classification accuracy, the benchmark dataset CIFAR-10 was employed, a further description of the dataset is presented in section 4.1. The second aim was to adopt the proposed framework to the planktonic domain and evaluate its performance on more complex datasets. For this purpose, three different datasets from the planktonic domain were employed. One of these included a dataset collected with the SINTEF developed SilCam employed in the AILARON project, hence an important measure for classification performance. All the datasets from the planktonic domain are presented in section 4.2.

4.1 CIFAR

The **CIFAR-10** (Canadian Institute For Advanced Research) [36] dataset consist of 60,000 32x32 colour images uniformly divided into 10 classes. It is commonly used as a benchmark dataset to train and test machine learning algorithms and is the most frequently tested dataset in the field of active learning [41]. The CIFAR-10 dataset is applied in experiments in this thesis to have a common ground for validating AL methods proposed in the literature and compare them with the proposed CIRAL framework. Example samples from the dataset are illustrated in figure 4.1 and a visualization of the feature space is presented in figure 4.2.

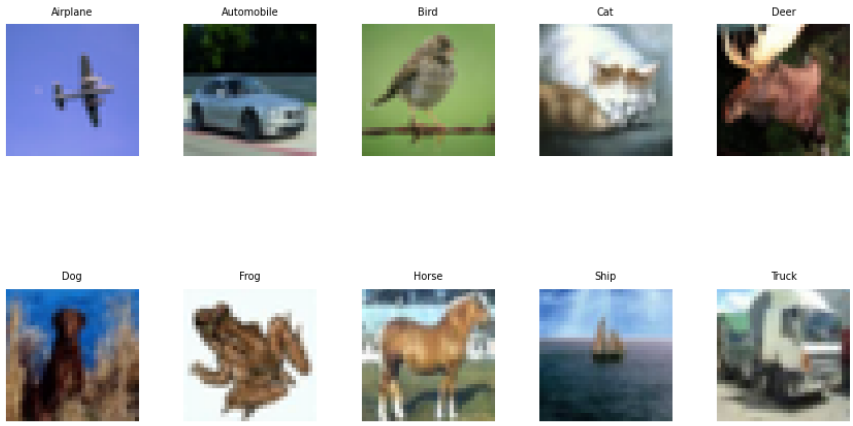


Figure 4.1: Samples from the CIFAR-10 dataset.

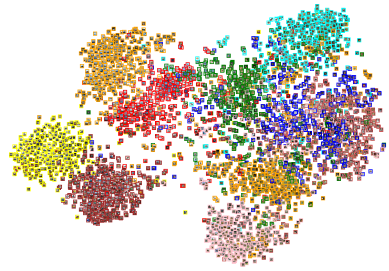


Figure 4.2: Feature visualization of the CIFAR-10 dataset. The different colored data points represent the images of the 10 different classes from the CIFAR dataset. With the T-SNE algorithm [58], the images are projected onto the two- dimensional feature space.

4.2 Plankton data

The main objective of this thesis is to apply active learning to the planktonic domain. Hence, it is essential to test and validate different methods on a variety of plankton datasets. To achieve this, three different datasets from the planktonic domain are selected, representing different levels of class balance, image quality, and image dimension.

4.2.1 AILARON

The **AILARON** dataset consists of planktonic data divided into six different classes. The collection of images was captured in the years between 2015 and 2018 in the fjord of Trondheim. An AUV with the SINTEF developed SilCam was employed to capture the images. Samples from the dataset are presented along with the class distribution in figure 4.3. A visualization of the feature space is presented in figure 5.3. The classes consist of four different categories of planktonic species, a category with air bubbles and one category of uncategorized images labeled 'other'. Comparing with the other plankton datasets in figure 4.5 and 4.7, one can observe that the AILARON dataset has a lower resolution and not equally as distinct features. The images do also, before transformation, have a greater variation in width and height. These variances make the application of the dataset for classification more difficult, but also more important in terms of validating different AL methods. However, the class distribution is fairly balanced compared to the Kaggle dataset presented in 4.2.2.

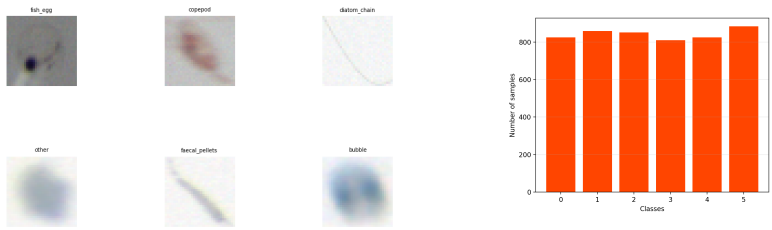


Figure 4.3: AILARON sample images and class distribution.

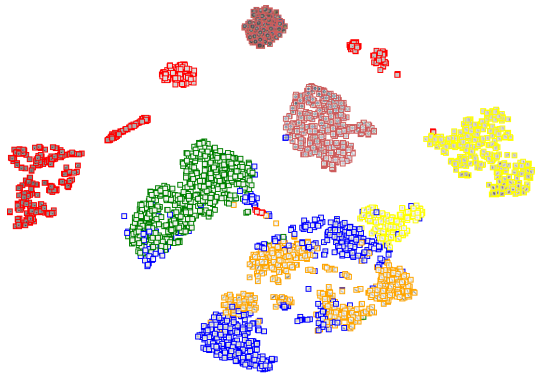


Figure 4.4: Feature visualization of the AILARON dataset. The different colored data points represent the images of the 6 different classes from the AILARON dataset. With the T-SNE algorithm [58], the images are projected onto the two- dimensional feature space.

4.2.2 Kaggle

The **Kaggle** plankton dataset [3] from the National Data Science Bowl (NDSB) of 2015 is a collection of images collected in the straits of Florida. A towed, underwater imaging system was employed to capture the images. Close to 50M plankton images got captured in the period May-June 2014, from which approximately 30,000 images have been labeled by Hatfield scientists. The images are ranging from a 30x30 dimension of the smallest up to 400x400 for the largest ones. Samples from the most prominent classes are provided together with the class distribution in figure 4.5. A visualization of the feature space is presented in figure 4.6. The images have high quality and cover a lot of different classes, and for this reason, the dataset is considered among the best possible of its kind [3]. It was initially published as a part of the NDSB competition and the winners, team *DeepSea*, reported a validation accuracy of 82%. Comparing with the findings of [14], reporting that domain experts can maintain $67 - 83\%$ self-consistency in taxa labeling, the supervised classification proved to compete with human domain experts in terms of classification accuracy.

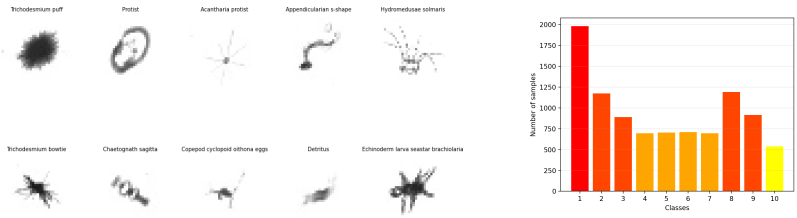


Figure 4.5: Kaggle sample images and class distribution.

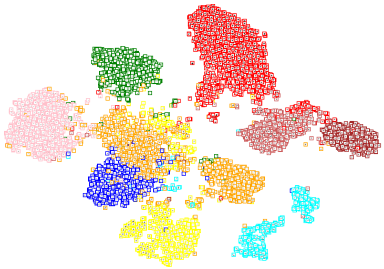


Figure 4.6: Feature visualization of the Kaggle dataset. The different colored data points represent the images of the 10 different classes from the Kaggle dataset. With the T-SNE algorithm [58], the images are projected onto the two- dimensional feature space.

While the Kaggle dataset has excellent image quality, the chosen subset of classes has a heavy imbalance towards the *"trichodesmium puff"* class as can be observed in 4.5, making it a good measure to see how well the different AL frameworks tackle the imbalance. To account for the imbalance in terms of evaluation methods, the balanced accuracy metric described in section 2.2.4 is employed.

4.2.3 Pastore

The **Pastore** dataset [45] is composed of 5000 plankton images evenly distributed over 10 different classes. The different freshwater planktonic species were captured on video by a lensless microscope as a part of the work with the annotation-free plankton

classifier approach described in chapter 3. As a part of the proposed pipeline, a series of the captured videos, each ten seconds long, are processed to generate cropped images of each plankter. Ten representative examples from the dataset are presented in figure 4.7 and a visualization of the feature space is presented in figure 4.8.

Figure 4.7: Pastore samples images.

The pastore dataset combines a balanced class distribution with excellent image quality, hence works as a good combination of the datasets described in 4.2.2 and 4.2.1.

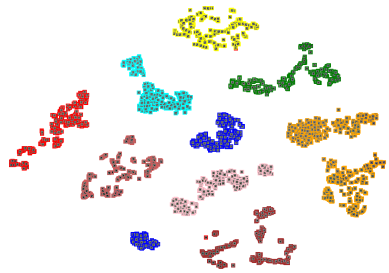


Figure 4.8: Feature visualization of the Pastore dataset. The different colored data points represent the images of the 10 different classes from the Pastore dataset. With the T-SNE algorithm [58], the images are projected onto the two- dimensional feature space.

4.3 Data pre-processing

The images in the described datasets appear in a multitude of different formats and dimensions. Since neural networks demand fixed input dimensions, as described in 2.2.1, some mild pre-processing is necessary to make the data readily available as input to the network. The pre-processing step employed consists of three parts, firstly the images are transformed into tensors, as described in section 2.2.1, optimizing the images for processing in the GPUs employed in the experiments. Secondly, the dimensions of the tensors are transformed to a fixed size due to the fixed dimensions of the filters in the learning network. Lastly, to make the training more robust and speed up the convergence, a normalization step is applied by transforming all the pixel values in the aforementioned tensors from the range (0,255) to (0,1).

Chapter 5

Methodology

The methodology chapter presents the framework for the novel active learning approach proposed in this thesis. It further describes the learning models employed in the framework together with applied data augmentation techniques. Lastly, a presentation of the implementation framework including relevant open source software, software packages, and hardware-specific details are included at the end of the chapter.

5.1 Proposed active learning framework

To improve on existing active learning frameworks presented in the literature, a novel method for active learning is proposed in this thesis. By leveraging research in the field of AL, this novel framework aims to combine the advantages of an informative and a representative query strategy in a hybrid active learning strategy. The framework described is the work proposed in [26] and presented on the "*13th International Conference on Digital Image Processing*" (ICDIP). Different from the presented work is the data augmentation module, described in section 5.3, that was added after the paper submission to increase the overall classification accuracy and in particular, give a performance enhancement for the informative metric employed. From figure 2.10, illustrating a batch of samples queried with an informative strategy, one can observe a lot of queries from the same area in the latent space, suggesting high correlation, and redundancy among the samples with regard to the information provided to the classifier. Based on this inefficiency in the sample querying, a representative metric is integrated into an informative-based active learning framework with a two-fold motivation. First, by initially incorporating the full feature space, it enables the image

classifier to utilize the full data distribution of the unlabeled samples, hence minimizing the unknown unknowns, i.e the knowledge gap, described in section 2.4.3. Moreover, it also avoids the redundant sampling previously described. This proposed hybridization enables querying of informative samples that also best represent the feature space of the unlabeled data. Figure 5.1 illustrates how the informative and representative metrics are combined through a trade-off function for hybridization of active learning. By initially presenting all samples for the representative metric, the learning model will gain an overview of the whole feature space. Further, as the labeled pool is incremented with new, queried samples in each AL cycle, the learning model will observe new features and update its inter-class decision boundaries correspondingly. These decision boundaries are represented by the trainable model parameters $\mathcal{A}_k$ in algorithm 3.

Algorithm 3 CIRAL: Combined informative and representative active learning extended with the augmentation module.

Require: Unlabeled samples D_0^U
Require: Initially labeled samples D_0^L
Require: Query budget **B**
Require: Batch size β
Require: Set of hyper-parameters to train the network $\mathcal{H}$
Require: Set of data augmentation techniques $\mathcal{T}$
  $D_k^L = D_0^L$
  $D_k^U = D_0^U$
  while $D_k^L - D_0^L \leq$ **B do**
    $\mathcal{A}_k = \textsc{Train}(D_k^L, \mathcal{H}, \mathcal{T})$
    for $x_i \in D_k^U$ **do**
      $r_i \leftarrow \textsc{DeepFool}(x_i, \mathcal{A}_k)$
    end for
    $b_i \longleftarrow \textsc{TradeOff}(r_i)$
    $Q_k \longleftarrow \textsc{MinMax}(b_i, \beta)$
    $D_{k+1}^L \longleftarrow D_k^L \cup Q_k$
    $D_{k+1}^U \longleftarrow D_k^U \backslash Q_k$
  end while

Still, as the training proceeds and the model become more confident, the decision boundaries become more static. Wherefore it becomes increasingly important to put weight on the samples that are in the proximity of the boundary to let the learning model fine-tune its decision boundaries. This switch from representative to informative

samples is achieved by employing the TradeOff method described in algorithm 4, which eventually ignores samples found at large distances away from the decision boundary.

Algorithm 4 TradeOff: Hybrid AL trade-off function.

Require: Ranked informative samples X_i
Require: Trade-off constant $K_K = 1$
Require: Trade-off rate $\delta \in (0, 1)$
$b_i \leftarrow X_i[0 : K_k]$
$K_{k+1} \leftarrow K_k \cdot \delta$
return b_i

To find the aforementioned distance, the DeepFool [42] algorithm, described in section 2.5.1, is employed in the proposed framework, to compute adversarial attacks to find a proxy for the distance to the decision boundary. By adding the above-mentioned data augmentation module, described in 5.3, to the framework, the network will improve its decision boundaries from training on more diverse samples, and resultingly improve the accuracy of the boundary distance found by the informative metric.

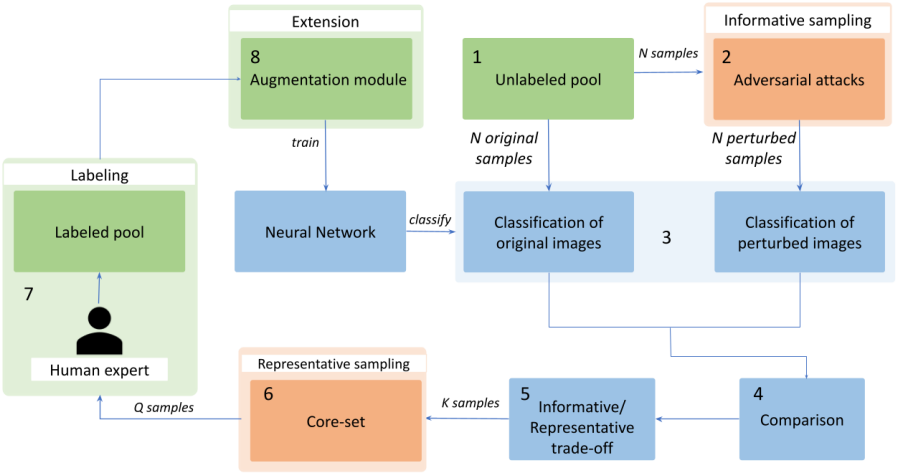


Figure 5.1: The proposed hybrid active learning framework. Combined Informative and Representative Active Learning (CIRAL).

To find the representative samples among the queried informative samples, the core-set approach described in 2.5.2 is employed, in particular, the MinMax problem presented in algorithm 2 is solved at each AL cycle. A number of samples, Q_k, from the representative sampling, is then queried to a human oracle for labeling and the AL cycle is then repeated. This process continues until a labeling budget **B** is exhausted. In summary, the aim behind this proposed hybridization has three folds

- The model will have a good initialization from incorporating the full feature space in the early rounds of querying and training.
- Adding representative sampling to the queried informative samples prevents redundant labeling representation from the same area of model uncertainty
- As the softmax layer in neural networks has shown to be a bad proxy for the uncertainty of neural network [24, 50, 52], an adversarial active learning method is employed. This method has previously shown good results [16], however it was not employed with sub-modular heuristics as is done in this thesis.

5.2 Employed image classifiers

Due to the different complexity in the datasets presented in chapter 4, two different image classifiers were employed in the experiments conducted for this thesis. For the CIFAR dataset, ResNet-18 was employed, whereas, for the planktonic datasets, a custom network was employed, both of which are presented in the upcoming sections. The optimization of the image classifiers was not in the scope of this thesis, however, to best validate the performance of the different active learning strategies, it was of importance to employ classifiers that performed well in terms of classification accuracy.

ResNet-18

Given enough capacity, a neural network with one hidden layer is sufficient for approximating any continuous function according to the universal approximation theorem [13]. However, as described in section 2.1, such massive layers lack flexibility and generalizability and are prone to overfitting the training data. To overcome this, novel neural networks have a modularized design that forms a hierarchical decomposition of the input image. This enables the network to decompose and interpret the different images based on their features. Nevertheless, even though more layers mean less weight on each layer and thus more flexibility, this type of deep architecture leads to problems known as *vanishing gradient problem* and *exploding gradient problem*.

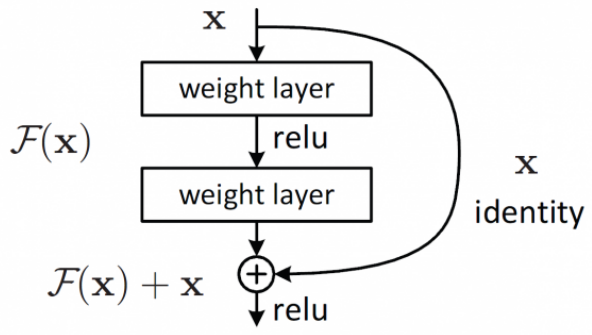


Figure 5.2: ResNet skip connection block (Illustrated by [27]).

The backpropagation described in 2.2.1 calculates the gradient of the loss function with regards to the weights in each layer. The gradient of the first layers in the network will then become either infinitely small or very big from the series of multiplication caused by the chain rule. That is, the gradient for the layer z with respect to the weight w_i in a neural network with depth j is

$$\frac{\partial z}{\partial w_i} = \sum_j \frac{\partial z}{\partial y_j} \frac{\partial y_j}{\partial w_i} \tag{5.1}$$

This in effect will, for infinitely small gradients, lead to very slow training of the early layers. To overcome this, [27] proposed a novel network design with skip connection blocks as illustrated in figure 5.2, which allows the model to backpropagate through the identity function, effectively preventing vanishing gradients. Further, instead of mapping the identity function $\mathcal{F}(x) = x$, the skip connection block allows for mapping of the zero-function $\mathcal{F}(x) = 0$, which is an easier and computational more efficient mapping. Formally, the underlying mapping from figure 5.2 becomes

$$\mathcal{H}(x) = \mathcal{F}(x) + x \tag{5.2}$$

a recasting gives the residual mapping

$$\mathcal{F}(x) = \mathcal{H}(x) - x \tag{5.3}$$

Consequently, $\mathcal{F}$(x) only needs to learn any change in x, i.e residual, from the underlying identity mapping, hence the name residual mapping. The details of the ResNet-18 architecture can be observed in table 5.1.

ResNet-18		
Layer type	**Output size**	**Layer details**
Convolution 1	112x112	7x7, 64, stride 2
Convolution 2	56x56	3x3 max pool, stride 2
		$\begin{bmatrix} 3x3, 64 \\ 3x3, 64 \end{bmatrix}$ x 2
Convolution 3	28x28	$\begin{bmatrix} 3x3, 128 \\ 3x3, 128 \end{bmatrix}$ x 2
Convolution 4	14x14	$\begin{bmatrix} 3x3, 256 \\ 3x3, 256 \end{bmatrix}$ x 2
Convolution 5	7x7	$\begin{bmatrix} 3x3, 512 \\ 3x3, 512 \end{bmatrix}$ x 2
	1x1	Average pool, 1000-d FC, softmax

Table 5.1: ResNet-18 architecture.

Custom network architecture

The network employed for the plankton datasets is described in table 5.2. It is consisting of only four convolutional layers and is considerably smaller in the number of trainable parameters compared to the previously described ResNet-18. However, as discussed in section 2.1, to avoid overfitting the dataset, it is important to limit the number of parameters in the learning network. The custom network proved to give good accuracy and avoid overfitting when tested on the planktonic dataset and was further employed in the experiments presented in chapter 6.

Table 5.2: Custom network architecture

Layer 1	Convolutional layer with 32 filters, kernel size=5
Layer 2	Convolutional layer with 32 filters, kernel size=5
	2x2 Max pooling
Layer 3	Convolutional layer with 64 filters, kernel size=5
	2x2 Max pooling
Layer 4	Fully-connected 1000-d layer
Layer 5	Fully-connected softmax layer with probability output

5.3 Data augmentation

A data augmentation module was added to the AL framework with a two-fold motivation. First, to increase the robustness of the framework and ensure its adaptability to the planktonic domain. Second, by adding the augmentation module, the classifier performance increases, resulting in better decision boundaries for the informative metric to work with. Regularization techniques are often disregarded in AL research [41] since it is considered to only scale the existing relative performance of different AL strategies. However, in this thesis, the regularization step is considered to be benefiting the AL strategy which is dependent on good decision boundaries, hence improving its performance relative to other methods.

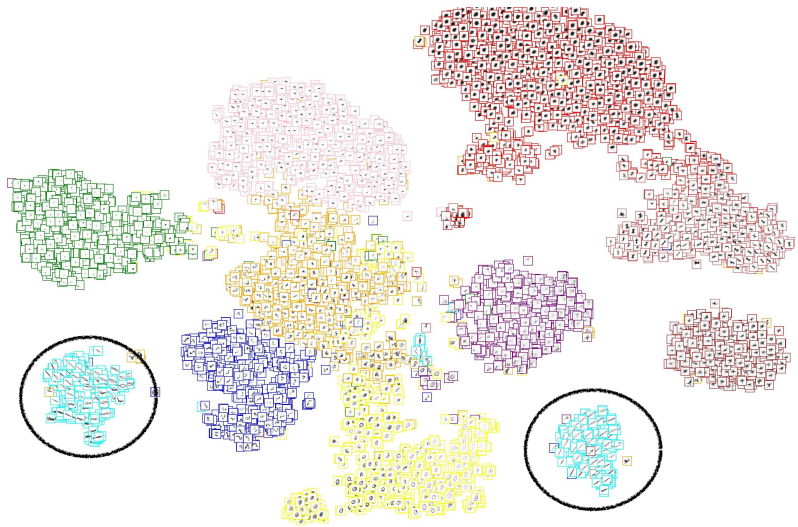


Figure 5.3: Visualization of the plankton classes from the Kaggle dataset shows how the "Chaetognath Sagitta" class is separated into two groups based on its orientation. The plankton images are projected onto the two-dimensional feature space using the T-SNE algorithm [58].

From an investigation of the feature clusters in figure 5.3, it was evident that the same class, with different internal orientations, are divided by the classifier into separate groups. Further, planktonic organisms are in general appearing in multiple different orientations as can be seen in figure 4.5, allowing for a heavy augmentation. Steps of augmentation applied in the proposed framework included

- **Horizontal flip**: Random with probability 0.5
- **Vertical flip**: Random with probability 0.5
- **Translation**: Random with shift 0.1,0.1 (pixel-wise)
- **Rotation**: Random with angle of 30°

The resulting data augmentation can be observed in figure 5.4 where a sample of images are plotted before and after the augmentation is applied. One can observe that the first and third samples from the right do not change much due to their initial form, hence a more aggressive data augmentation could be applied for those samples. This set of augmentation techniques are summarized as $\mathcal{T}$ in algorithm 3 and applied at run time to the images before they are fed into the learning network.

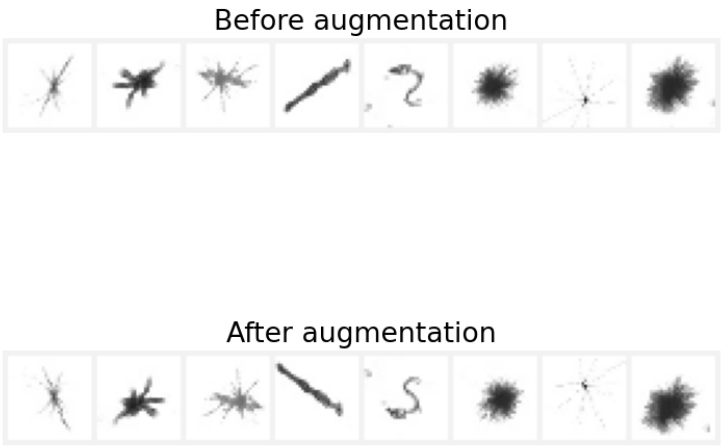


Figure 5.4: Comparison of samples before and after augmentation.

5.4 Implementation environment

The implementation of the ML and AL frameworks described in this thesis relied mainly on the use of *PyTorch*, which is an open-source machine learning library developed by the Facebook AI research lab [46]. At the granular level, PyTorch consists of several components allowing it to be used for high-speed numerical operations leveraging the power of GPUs and deep learning with high flexibility and speed. In particular, PyTorch is leveraging the use of tensors described in section 2.2.1 that accelerates computation when loaded into GPUs. PyTorch was chosen as the library for the implementation of the active learning strategies and neural networks because of its deep integration with Python and SciPy [60], the high level of support, and a minimal framework overhead. Relevant alternatives to PyTorch that were considered for implementation included TensorFlow and Caffe. Other relevant packages used in the implementation included the *Sklearn* library [47] and also *NumPy*, *Pandas* and *Matplotlib* from the SciPy ecosystem. These aforementioned packages are fundamental for scientific computing and visualizations with Python. The Python distribution platform Anaconda [4] was employed for package and dependency management. Furthermore, the T-SNE library [58] was used for the feature visualizations in chapter 4. All code for running the experiments described in this thesis is available out-of-the-box

at *https://github.com/AILARON/active-learning*. The experiments were conducted on a computer belonging to the AILARON project. The specifications of the computer can be observed in table 5.3. Importantly, all of the experiments ran in a Linux environment on the described GPUs, for which the aforementioned PyTorch library is optimized.

Computer specifications	
OS:	Ubuntu 18.04.3
CU:	Intel Core i9 - 9900K
GPU:	2x ASUS RTX2080Ti Turbo
RAM:	64 GB
SSD:	Crucial MX500 2TB

Table 5.3: Computer specifications.

Chapter 6

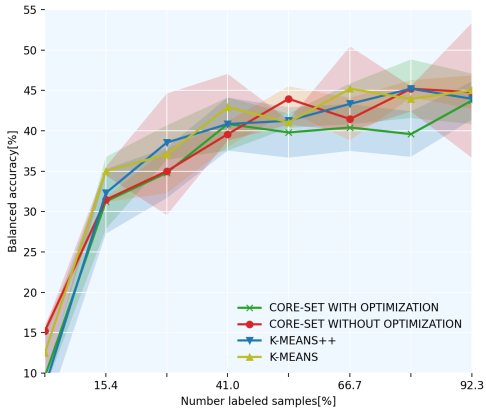
Experiments and results

This chapter presents the experiments conducted in this thesis together with visualizations and a brief analysis of the experimental results. All the experiments are conducted on the datasets described in chapter 4. The setup for the conducted experiments is as follows; An initial pool of 100 labeled data samples is selected to warm up the employed image classifier and a budget of at least 70% of the unlabeled samples is employed for each experiment. A conventional AL setup includes a human oracle in the loop who labels the queried samples, as described in section 2.4.1. Since this setup is impractical for the experiments conducted in this thesis, the human oracle was simulated by employing pre-labeled samples. All reported results are the mean of three trials and are presented with an uncertainty representing the standard deviation of the results. The evaluation measure employed is noted on the y-axis of the figure and is further described in section 2.2.4.

The next section will provide an evaluation of different representative metrics before a comparison of different AL strategies is conducted. Lastly, an experiment on the effect of data augmentation on AL strategies is presented in section 5.3.

6.1 Comparing representative metrics

This section presents visualizations and a brief analysis of the results from the experiments on different representative metrics. The aim of the experiments was to identify an applicable representative strategy for the proposed hybrid framework. In particular, the core-set and k-means approaches with different configurations as described in section 2.5.2 were employed in the experiments.

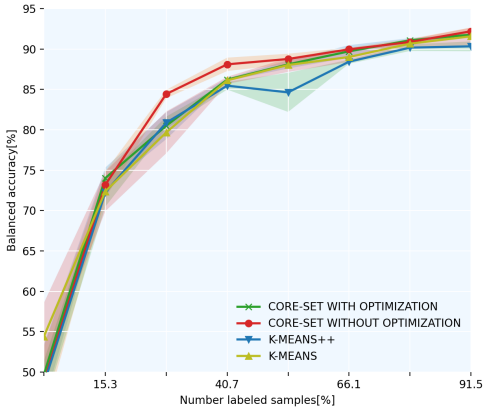


Method	Time
Core-set	00:03:52
Core-set w/opt	**00:30:09**
K-means	00:03:23
K-means++	00:03:40

Figure 6.1: (LHS) Comparison of accuracy for different representative strategies on the CIFAR-10 dataset. (RHS) The corresponding time consumption for each strategy.

Figure 6.1 and 6.2 visualize the results from the experiments conducted on the CIFAR-10 and AILARON datasets, respectively. A subset of 5000 images was randomly selected to represent the CIFAR-10 dataset, whereas the AILARON dataset consisted of 4840 images. The datasets got split into 80% unlabeled training samples and 20% testing samples. A query size of 500 was used in both experiments. The experimental results show marginal differences in classification accuracy, however, the core-set approach including an optimization module is prominent with high computational time. Further, minimal differences are apparent in the comparison of the k-means and k-means++ approaches, however, with a slight increase in computation time for the latter. One can observe more uncertainty in the results from the experiments on the CIFAR-10 dataset compared with the experiments on the AILARON dataset. Observing that the feature space of the Kaggle dataset illustrated in figure 4.6 is much more intertwined compared to the AILARON feature space visualized in figure 4.4. This observation suggests that the representative metrics benefit from a separated class distribution as illustrated in figure 4.4. It can be seen in figure 6.1 that the k-means approach has a much higher uncertainty compared to the k-means++ approach. As described in section 2.5.2, the k-means++ approach trades off a high computation time to gain a better initialization, and as result, a more stable clustering performance. As a result of the experiments presented, in particular, the experiment on the AILARON dataset, the core-set approach without the optimization module was selected for the AL framework

proposed in this thesis.



Method	Time
Core-set	00:03:13
Core-set w/opt	**00:48:30**
K-means	00:02:45
K-means++	00:03:48

Figure 6.2: (LHS) Comparison of accuracy for different representative strategies on the AILARON dataset. (RHS) The corresponding time consumption for each strategy.

6.2 Comparing active learning frameworks

This section presents visualizations and a brief analysis of the results from the experiments comparing different AL approaches. The approaches employed in the experiments are representing the broad categories of AL presented in chapter 2 and are compared with the proposed CIRAL framework described in chapter 5. A summary of the strategies is presented below.

- **Active learning by learning (ALL):** A hybrid approach, described in section 2.5.3, which varies between an informative and a representative method based on their predicted performance.
- **BADGE:** A hybrid approach, described in section 2.5.3, that incorporates uncertainty and diversity based on the gradients from the loss function.
- **Core-set:** A representative approach, described in section 2.5.2, that aims to represent the full data distribution by finding samples that cover the full feature space.
- **DFAL:** An informative approach, described in section 2.5.1, that uses adversarial attacks to calculate image perturbations as a proxy for the images' distance to

the nearest decision boundary.
- **CIRAL:** The proposed hybrid method, described in 5.1, combines a distance-based informative approach with a clustering-based representative strategy through a trade-off function.
- **Random:** Chooses at random a set of points for each round. Considered the benchmark for active learning methods. Also referred to as RBS (Random Benchmark Sampling).

For each dataset, two experiments are conducted with two different query sizes to measure the impact of the batch size on the classification accuracy. The query size is set to 200 and 400 for the CIFAR-10, Kaggle, and AILARON datasets, whereas 100 and 50 for the Pastore dataset. For each experiment, a bar plot is presented showing the distribution of the queried classes when approximately 50% of the budget has been queried. Each class is represented with the class number, and the height of the bar represents the particular classes' share out of all queried samples. The plot is added to illustrate how the different strategies prioritize in their sampling. The coming subsections motivate each dataset employed and presents the results from the experiments conducted.

6.2.1 Experiments on the CIFAR dataset

The CIFAR-10 dataset, described in section 4.1, was employed in the experiments to validate and benchmark different methods of active learning. A subset of 5000 samples was randomly selected from the full dataset. This training set got split into a training and testing set consisting of 4000 and 1000 samples, respectively. 200 of the samples from the testing set were further selected as a validation set. The next section will present visualizations and a brief analysis of the experiments conducted.

The results presented in figure 6.3b visualize how the CIRAL algorithm performs compared to informative and representative approaches, whereas figure 6.3a visualize the performance compared to other hybrid approaches, a query size of 200 was employed for both experiments. A full overview of the results can be observed in table 6.1. It is easily seen that none of the approaches are prominent in terms of classification performance and the results include a lot of uncertainty. It is to be noted that the CIFAR-10 dataset represents complex images with a lot of information, hence 4000 images is not enough to get an acceptable accuracy on the testing set. In figure 6.3b one can observe the random benchmark sampling (RBS), to perform on par or better than the other AL approaches. With the aforementioned complexity of the CIFAR-10 dataset, it can indicate that random sampling is an equally effective sampling technique as the other strategies when most of the training examples are considered valuable. However, when increasing the query size to 400 as is done for the experiments pre-

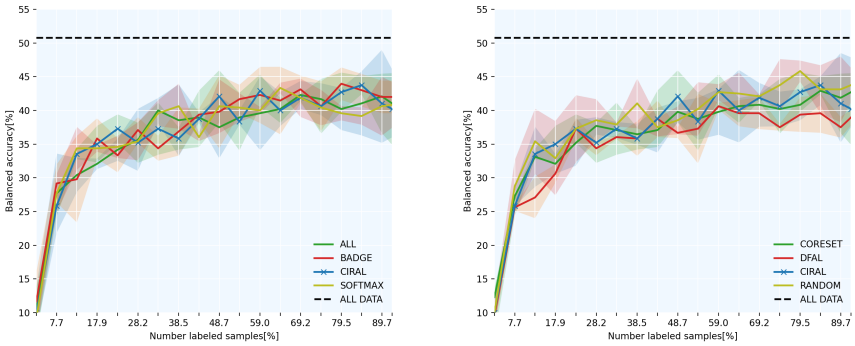
Method\Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
ALL	0.106	0.277	0.304	0.320	0.341	0.356	**0.399**	0.385	**0.389**	0.375	0.389	0.395	0.402	0.422	0.416	0.402	0.410	0.420	0.410
BADGE	0.116	**0.2916**	0.297	**0.358**	0.333	0.370	0.343	0.368	0.393	0.397	**0.416**	0.422	0.414	**0.431**	0.406	0.439	0.430	0.420	0.420
CIRAL	0.100	0.258	0.335	0.350	**0.372**	0.352	0.372	0.358	0.387	**0.420**	0.383	**0.429**	0.399	0.418	0.406	0.427	**0.437**	0.410	0.402
CORESET	**0.122**	0.272	0.331	0.320	0.352	0.377	0.370	0.364	0.370	0.397	0.387	0.397	0.406	0.408	0.402	0.408	0.429	0.418	0.427
DFAL	0.095	0.256	0.270	0.306	0.372	0.343	0.360	0.358	0.387	0.366	0.372	0.406	0.395	0.395	0.375	0.393	0.395	0.375	0.389
SOFTMAX HYBRID	0.091	0.277	0.343	0.343	0.345	0.354	0.395	0.406	0.360	0.406	0.404	0.399	**0.433**	0.418	0.404	0.395	0.391	0.406	0.408
RANDOM	0.100	0.287	**0.354**	0.329	0.372	**0.385**	0.379	**0.410**	0.375	0.385	0.402	0.427	0.425	0.420	**0.437**	**0.458**	0.431	**0.431**	**0.437**

Table 6.1: Balanced accuracy score from the experiment on the CIFAR dataset with a query size of 200. The best score in each round is marked with a bold font.

Method\Round	1	2	3	4	5	6	7	8	9
ALL	0.091	0.306	0.327	0.350	0.352	0.383	0.431	0.387	0.381
BADGE	0.088	0.322	0.367	0.392	0.392	0.399	0.408	0.413	0.413
CIRAL	0.091	0.310	0.383	0.398	**0.406**	0.406	0.410	0.384	0.410
CORESET	0.083	**0.353**	0.373	**0.393**	0.397	0.397	0.416	0.424	0.417
DFAL	0.091	0.310	0.360	0.379	0.404	**0.430**	0.441	0.439	**0.446**
SOFTMAX HYBRID	0.100	0.339	**0.391**	0.405	0.400	0.427	**0.446**	0.437	0.424
RANDOM	**0.094**	0.298	0.363	0.366	0.391	0.409	0.425	**0.440**	0.432

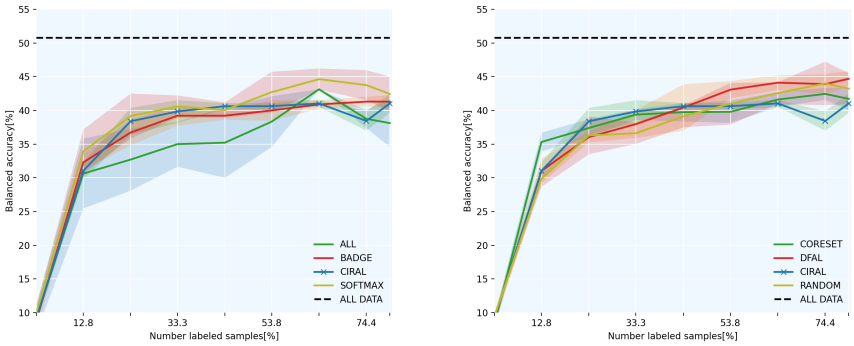
Table 6.2: Balanced accuracy score from the experiment conducted on the CIFAR dataset with a query size of 400. The best score in each round is marked with a bold font.

sented in figure 6.4b and 6.4a, the RBS performs worst. It can be observed in figure 6.4b that RBS needs approximately twice as many samples as the CIRAL approach to reach the same accuracy. The CIRAL approach shows, for a query size of 400, to be a good combination of the CORESET and DFAL approaches. While the DFAL has low performance in the beginning and the CORESET has a low performance at the end, the CIRAL has stable high performance. However, the results from later queries indicate that the CIRAL approach is unable to query the most informative samples, leading to worse performance than the DFAL approach. The CIRAL approach proves, however, to give a consistently high performance as can be seen in figure 6.4a and 6.4b, and is performing on par or better than the other hybrid methods up until 50% of the samples have been labeled.



(a) Hybrid approaches. (b) Informative/Representative approaches.

Figure 6.3: Result from comparison of approaches conducted on the CIFAR dataset with a query size of 200.

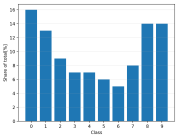


(a) Hybrid approaches. (b) Informative/Representative approaches.

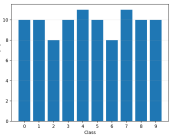
Figure 6.4: Result from comparison of approaches conducted on the CIFAR dataset with a query size of 400.

Figure 6.5 visualize the class distribution of the queried samples. The distributions show no particular difference apart from the unbalanced class distribution created by

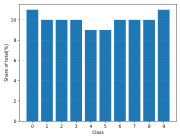
the DFAL and ALL strategy. In particular the ALL strategy suffers from this unbalanced query distribution as can be observed in figure 6.4a. It is apparent that a lot of the samples have been queried from the same area of uncertainty, hence the model is not provided with enough diverse features to give a good initial performance. This is especially prominent when compared to the hybrid methods which incorporate the full feature space in the early AL cycles. A full overview of the results from the experiments with a query size of 400 can be seen in table 6.2.



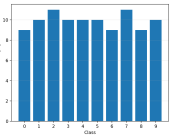
(a) Distribution of ALL.



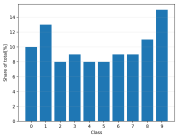
(b) Distribution of BADGE.



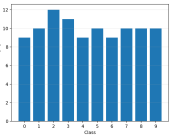
(c) Distribution of CIRAL.



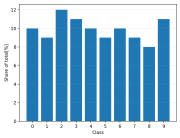
(d) Distribution of CORESET.



(e) Distribution of DFAL.



(f) Distribution of RANDOM.



(g) Distribution of SOFTMAX HYBRID.

Figure 6.5: Class distribution of the queried samples from the experiment conducted on the CIFAR dataset with a query size of 400.

6.2.2 Experiments on the AILARON dataset

The AILARON dataset described in section 4.2.1 was employed in the experiments due to its similarity with other datasets constructed for the AILARON project, hence a good benchmark for the different AL frameworks. The dataset consisting of 4840 images was split into a training, test, and validation set with 3933, 807, and 100 images, respectively. The custom network described in section 5.2 was employed as an image classifier for the experiments conducted.

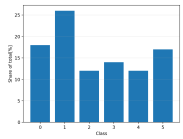
Method\Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ALL	0.504	0.646	0.748	0.795	0.835	0.852	0.867	0.884	0.895	0.902	0.909	0.913	**0.919**	**0.918**	**0.925**	0.914
BADGE	0.528	0.680	**0.795**	**0.813**	0.843	**0.859**	0.869	0.883	0.891	0.898	0.898	0.903	0.918	0.907	0.917	0.921
CIRAL	0.512	0.674	0.764	0.797	0.840	0.851	0.862	0.863	0.876	0.886	0.893	0.892	0.893	0.902	0.899	0.911
CORESET	0.499	0.703	0.769	0.805	0.836	0.847	0.869	0.881	0.884	0.892	0.903	0.909	0.908	0.907	0.914	0.916
DFAL	0.488	0.606	0.743	0.793	0.828	0.831	0.841	0.860	0.869	0.886	0.897	0.900	0.909	0.912	0.921	0.912
SOFTMAX HYBRID	0.3915	0.647	0.750	0.792	0.840	0.852	0.860	0.864	0.873	0.885	0.888	0.891	0.886	0.898	0.899	0.903
RANDOM	**0.534**	**0.681**	0.771	0.806	**0.845**	0.857	**0.879**	**0.893**	**0.899**	**0.904**	**0.912**	**0.916**	0.913	0.917	0.921	**0.929**

Table 6.3: Balanced accuracy score from the experiment on the AILARON dataset with a query size of 200. The best score in each round is marked with a bold font.

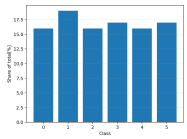
Method\Round	1	2	3	4	5	6	7	8	9
ALL	0.437	0.693	0.805	0.838	0.863	0.871	0.881	0.900	0.902
BADGE	**0.514**	0.729	0.799	0.839	0.871	0.884	0.897	0.893	0.903
CIRAL	0.464	0.701	0.801	0.844	0.868	0.883	0.894	0.907	0.913
CORESET	0.433	**0.747**	0.793	0.833	0.858	0.881	0.882	0.895	0.907
DFAL	0.510	0.702	0.811	0.835	0.868	0.882	0.900	**0.912**	0.913
SOFTMAX HYBRID	0.492	0.715	**0.813**	**0.850**	**0.876**	**0.897**	**0.901**	0.905	**0.914**
RANDOM	0.474	0.712	0.803	0.847	0.875	0.883	0.894	0.902	0.909

Table 6.4: Balanced accuracy score from the experiment on the AILARON dataset with a query size of 400. The best score in each round is marked with a bold font.

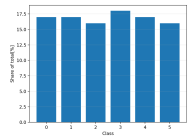
Comparing the results of the experiments with query size of 200 and 400, shown in table 6.3 and 6.4, one can observe that the general accuracy drops when the query size is increased. This is in contrast to what could be observed for the experiment on the CIFAR dataset in section 6.2.1. Further, it is prominent in figure 6.7a that the different methods in the experiment with a query size of 200 are performing equally well, however with ALL and BADGE performing marginally better than the other hybrid methods. In figure 6.7b one can see that RBS achieves the best performance, however, when the query size is doubled as seen in figure 6.8b, the CIRAL approach is performing on par or better than RBS. Equal performance for the different hybrid methods can also be observed in figure 6.8a. This can be explained by the larger batch size which gives the different strategies increased probability of capturing relevant information in the queries.



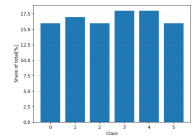
(a) Distribution of ALL.



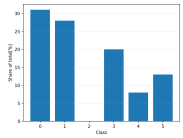
(b) Distribution of BADGE.



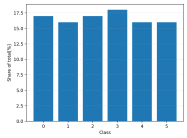
(c) Distribution of CIRAL.



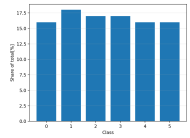
(d) Distribution of CORESET.



(e) Distribution of DFAL.

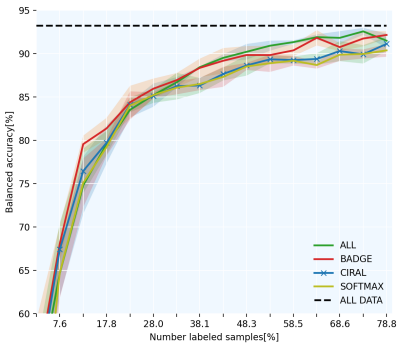


(f) Distribution of RANDOM.

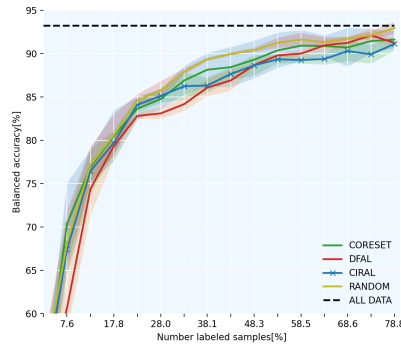


(g) Distribution of SOFTMAX HYBRID.

Figure 6.6: Class distribution of the queried samples from the experiment conducted on the AILARON dataset with a query size of 400.

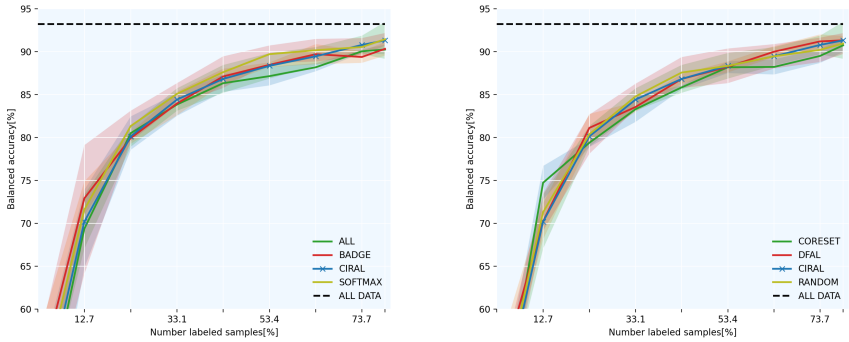


(a) Hybrid approaches.



(b) Informative/Representative approaches.

Figure 6.7: Result from comparison of approaches conducted on the AILARON dataset with a query size of 200.



(a) Hybrid approaches.

(b) Informative/Representative approaches.

Figure 6.8: Result from comparison of approaches conducted on the AILARON dataset with a query size of 400.

When comparing the class distribution of the queried samples illustrated in figure 6.6, one can observe that DFAL and ALL are prominent in terms of unbalanced query distribution. An interesting observation is that the DFAL method has not queried any data samples from 'Class 3' when half of the budget has been queried. Still, it achieves a good performance suggesting that the data incorporated in the initial labeled samples provided enough information to correctly classify the samples from this category. From table 6.3, it is evident that RBS are performing best on the AILARON dataset with a query size of 200. However, as seen in table 6.4, the Softmax Hybrid is outperforming the other approaches for a query size of 400.

6.2.3 Experiments on the Kaggle dataset

The Kaggle dataset described in section 4.2.2 was employed in the experiments to represent an unbalanced planktonic dataset, as illustrated in 4.5. For plankton classification, the most realistic scenario is to have a few dense classes and multiple sparse numbered classes. Hence, it is desirable to have an active learning strategy proven to work well with unbalanced class distributions. The dataset consists of 9379 images split into a training, testing, and validation set with 7484, 1516, and 379 images, respectively. The results from the experiments on the Kaggle dataset is shown in figure 6.5 and 6.6 for a query size of 200 and 400, respectively. Interesting to observe from the plotted results in figure 6.9, is how the ALL and DFAL methods are performing significantly better than the other methods from the point when approximately 20% of the samples have

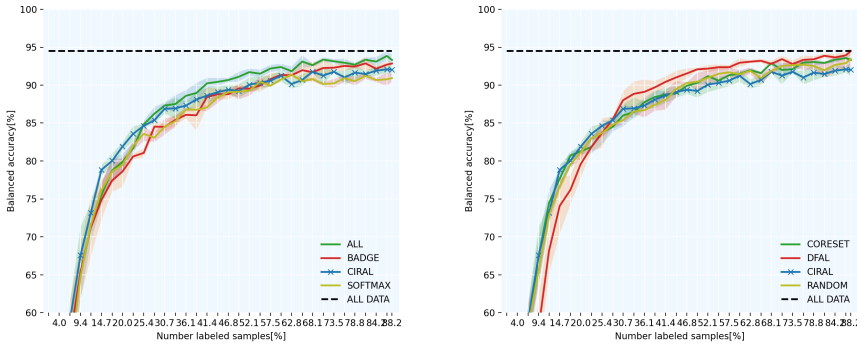
been labeled. This observation suggests that they are better at identifying informative samples at later query cycles compared with the other methods in the experiment. One can also observe in figure 6.9b, that the proposed CIRAL frameworks benefit from this and are performing better than RBS up until 40% of the samples have been labeled. From that point, only a minor increase is achieved with twice the amount of labeled samples. This trend is also prominent when comparing with the other hybrid methods in figure 6.9a. When increasing the query size from 200 to 400, the differences in the results from the previous experiments are even more apparent. Observing from figure 6.10b how the DFAL approach suffers from not incorporating the full feature space in the early training round, but performs significantly better in the later rounds. Moreover, the CIRAL approach can have a good initial performance and in general outperform the core-set approach due to the incorporated informative metric. However, it lacks the utilization of informative samples in which the DFAL method is better at identifying. This challenge will be further elaborated in the discussion in chapter 7.

Method\Round	**1**	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**	**21**	**23**	**25**	**27**	**29**	**31**	**33**
ALL	0.234	0.553	0.712	0.755	0.798	0.848	0.873	0.885	0.902	0.906	0.917	0.922	0.918	0.926	0.931	0.929	0.933
BADGE	0.221	0.565	0.712	0.774	0.805	0.845	0.854	0.860	0.884	0.888	0.895	0.899	0.913	0.919	0.922	0.925	0.928
CIRAL	0.243	**0.589**	0.731	0.800	**0.835**	0.853	0.869	0.881	0.890	0.892	0.903	0.912	0.906	0.912	0.910	0.914	0.920
CORESET	0.215	0.578	**0.744**	**0.807**	0.818	0.845	0.864	0.884	0.889	0.903	0.906	0.914	0.915	0.919	0.929	0.929	0.935
DFAL	0.238	0.464	0.681	0.761	0.818	**0.855**	**0.888**	**0.897**	**0.910**	**0.920**	**0.923**	**0.929**	**0.932**	**0.934**	**0.933**	**0.938**	**0.939**
SOFTMX HYBRID	0.237	0.537	0.715	0.786	0.820	0.830	0.852	0.867	0.883	0.891	0.904	0.907	0.905	0.901	0.909	0.912	0.907
RANDOM	**0.285**	0.579	0.728	0.795	0.829	0.847	0.865	0.874	0.894	0.904	0.914	0.914	0.909	0.925	0.926	0.919	0.928

Table 6.5: Balanced accuracy score from the experiment on the Kaggle dataset conducted with a query size of 200. The best score in each round is marked with a bold font.

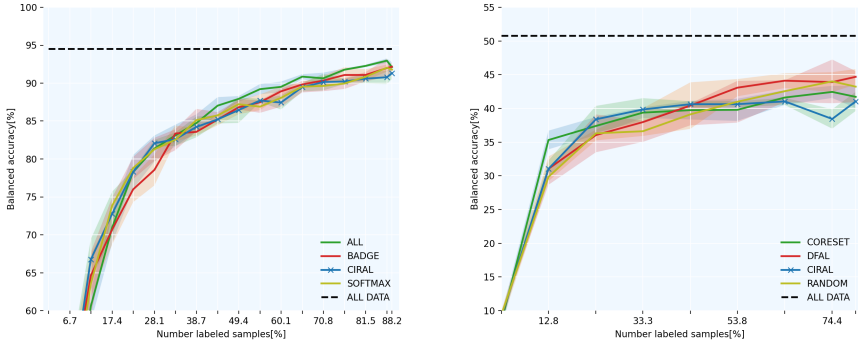
Method\Round	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
ALL	0.213	0.416	0.605	0.709	0.782	0.813	0.831	0.847	**0.870**	0.879	0.892	0.895	0.908	0.906	0.917	0.922	**0.929**	0.921
BADGE	0.204	0.484	0.646	0.706	0.759	0.785	**0.833**	0.835	0.851	0.868	0.874	0.889	0.898	0.903	0.910	0.910	0.919	0.921
CIRAL	0.236	0.484	**0.667**	0.727	0.782	**0.820**	0.826	0.842	0.852	0.864	0.877	0.874	0.894	0.901	0.902	0.905	0.907	0.913
CORESET	0.265	**0.508**	0.647	0.727	0.770	0.807	0.817	0.834	0.845	0.855	0.866	0.877	0.882	0.899	0.907	0.911	0.909	0.908
DFAL	**0.273**	0.388	0.513	0.631	0.717	0.769	0.827	**0.854**	0.865	**0.887**	**0.907**	**0.911**	**0.909**	**0.918**	**0.920**	**0.923**	0.928	**0.930**
SOFTMAX HYBRID	0.229	0.460	0.635	**0.738**	**0.788**	0.812	0.825	0.851	0.857	0.872	0.868	0.880	0.895	0.896	0.899	0.908	0.920	0.917
RANDOM	0.255	0.433	0.622	0.727	0.771	0.807	0.821	0.837	0.855	0.873	0.886	0.884	0.888	0.902	0.900	0.909	0.919	0.912

Table 6.6: Balanced accuracy score from the experiment on the Kaggle dataset conducted with a query size of 400. The best score in each round is marked with a bold font.



(a) Hybrid approaches. (b) Informative/Representative approaches.

Figure 6.9: Result from comparison of approaches conducted on the Kaggle dataset with a query size of 200.

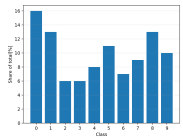


(a) Hybrid approaches. (b) Informative/Representative approaches.

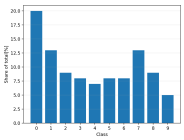
Figure 6.10: Result from comparison of approaches conducted on the Kaggle dataset with a query size of 400.

Comparing the results of the class distribution of the queried samples from figure 6.11, one can observe that the ALL and DFAL approaches are tackling the class imbal-

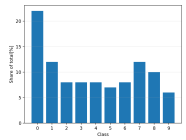
ance well and can query samples from the less numerous classes. It is evident that the other hybrid and representative methods are too focused on the most prominent class in comparison and end up with more redundant samples. Comparing this result with the plotted accuracy, one can observe that the DFAL and ALL approaches are also the approaches that are performing best.



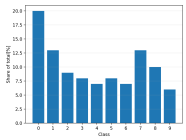
(a) Distribution of ALL.



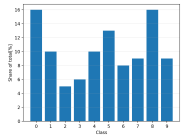
(b) Distribution of BADGE.



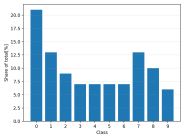
(c) Distribution of CIRAL.



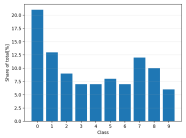
(d) Distribution of CORESET.



(e) Distribution of DFAL.



(f) Distribution of RANDOM.



(g) [Distribution of SOFTMAX HYBRID.

Figure 6.11: Class distribution of the queried samples from the experiment conducted on the Kaggle dataset with a query size of 400.

6.2.4 Experiments on the Pastore dataset

The Pastore dataset described in section 4.2.3 was employed in the experiments to represent an optimal dataset, combining high image quality with a balanced class distribution. Interesting observations from this experiment were expected to be which AL strategy that would perform best in a low-data setting, and which strategy would utilize the possible redundant information in the dataset best. The Pastore dataset consists of 5000 samples divided into training, testing, and validation set of 4000, 800, and 200 samples, respectively. Observing from the results in figure 6.13 that a lot of redundant information is incorporated in the Pastore dataset. The first experiment was conducted with a query size of 200, with the results presented in table 6.7. An observation to be made for this dataset is how the AL strategies benefit from the use of representative metrics instead of informative metrics. One can observe in figure 6.13 how the ALL and DFAL methods fall short compared to the other approaches. This is the opposite of the results from the experiments on the Kaggle dataset presented in section

6.2.3. To get a more clear picture of the performance of the different approaches on the Pastore dataset, the query size and budget were lowered to 50 and 500, respectively. The results from the new experiments is shown in figure 6.14. The pattern from the previous experiment is still evident, however, the proposed CIRAL framework proves to give the best performance out of the compared approaches in this low-data setting. Interestingly, the same informative metric is used for both DFAL and CIRAL, however, the CIRAL method employs a sub-modular heuristic which it benefits from in this particular experiment.

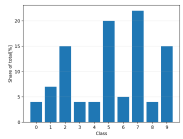
Method\Round	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
ALL	0.134	0.368	0.614	0.810	0.909	0.950	0.971	0.980	0.985	**0.996**	0.996	**0.998**	**0.997**	0.997	0.997	0.998
BADGE	0.101	0.385	0.688	0.902	0.957	0.974	0.985	0.988	0.991	0.992	0.996	0.993	0.994	0.996	0.997	0.997
CIRAL	0.129	0.388	0.775	**0.916**	0.963	0.980	0.985	**0.989**	**0.995**	0.995	**0.997**	0.996	0.995	0.998	0.997	0.997
DFAL	0.100	0.323	0.643	0.809	0.897	0.934	0.967	0.968	0.979	0.9840	0.992	0.994	0.996	0.990	0.994	0.997
CORESET	**0.139**	**0.393**	**0.800**	0.907	**0.965**	**0.981**	0.983	0.989	0.990	0.994	0.995	0.995	0.995	0.996	**0.998**	0.996
SOFTMAX HYBRID	0.099	0.370	0.772	0.906	0.949	0.971	0.981	0.985	0.987	0.990	0.992	0.994	0.996	0.996	0.996	0.997
RANDOM	0.119	0.417	0.770	0.914	0.962	0.981	**0.986**	0.993	0.994	0.995	0.996	0.997	0.997	**0.999**	0.997	**0.999**

Table 6.7: Balanced accuracy score from the experiment on the Pastore dataset conducted with a query size of 200. The best score in each round is marked with a bold font.

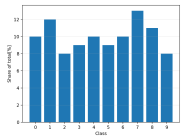
Comparing the distribution of queried samples in figure 6.12, it is prominent how the CIRAL, ALL, and DFAL approaches have a more unbalanced query distribution compared to the other approaches. An interesting observation to be made is how the CIRAL approach has the majority of queried samples from 'Class 8', identical to ALL and DFAL. However, the CIRAL is more balanced in its query from other classes and as can be observed in table 6.7 and 6.8, it performs better or on par with the other strategies for the experiments on the Pastore dataset.

Method\Round	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
ALL	0.131	0.204	0.324	0.462	0.670	0.773	0.855	0.884	0.916	0.942	0.955
BADGE	0.130	0.234	0.440	0.610	0.788	0.844	0.896	0.935	0.958	**0.973**	0.980
CIRAL	**0.137**	**0.298**	**0.461**	**0.637**	0.811	**0.892**	0.917	**0.949**	0.962	0.972	**0.980**
CORESET	0.099	0.231	0.373	0.612	0.779	0.860	0.930	0.942	**0.968**	0.965	0.976
DFAL	0.108	0.237	0.417	0.590	0.725	0.816	0.844	0.864	0.875	0.898	0.931
SOFTMAX HYBRID	0.115	0.227	0.429	0.631	0.760	0.854	0.919	0.939	0.956	0.71	0.973
RANDOM	0.099	0.265	0.459	0.625	**0.813**	0.880	**0.920**	0.940	0.960	0.970	0.980

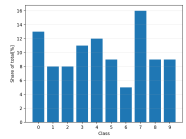
Table 6.8: Balanced accuracy score from the experiment on the Pastore dataset conducted with a query size of 50. The best score in each round is marked with a bold font.



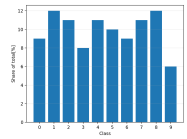
(a) Distribution of ALL.



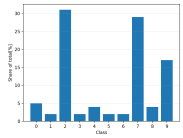
(b) Distribution of BADGE.



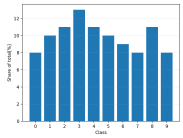
(c) Distribution of CIRAL.



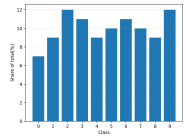
(d) Distribution of CORESET.



(e) Distribution of DFAL.

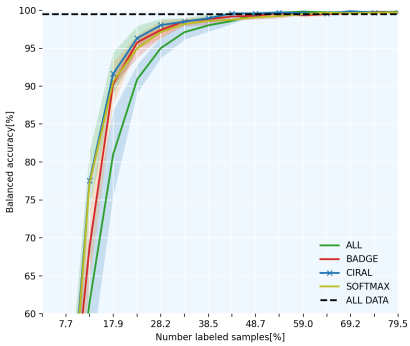


(f) Distribution of RANDOM.

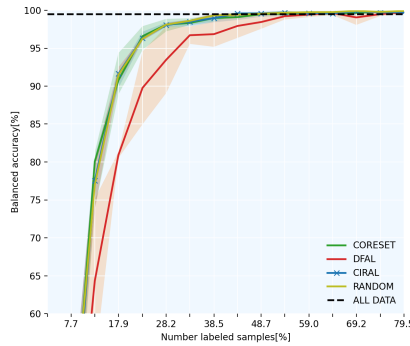


(g) Distribution of SOFTMAX HYBRID.

Figure 6.12: Class distribution of the queried samples from the experiment conducted on the Pastore dataset with a query size of 50.

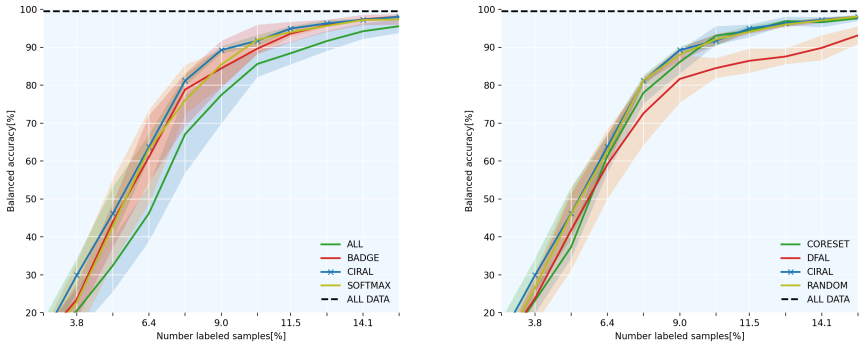


(a) Hybrid approaches.



(b) Informative/Representative approaches.

Figure 6.13: Result from comparison of approaches conducted on the Pastore dataset with a query size of 200



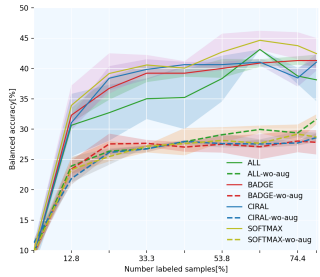
(a) Hybrid approaches.

(b) Informative/Representative approaches.

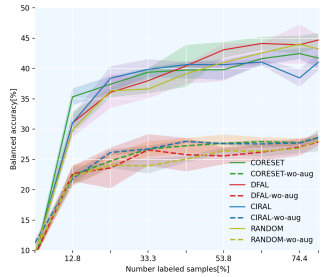
Figure 6.14: Result from comparison of approaches conducted on the Pastore dataset with a query size of 50.

6.3 Experiments on the effect of data augmentation

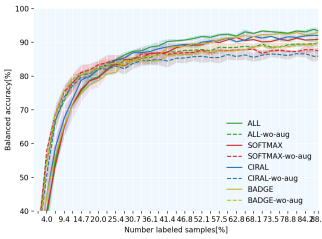
Comparing the results of the approaches with and without the augmentation module in figure 6.15, it is evident that the module is beneficial in terms of classification performance. Yet, an interesting observation is that the non-augmented approaches have an initial better performance than the augmented approaches. However, the augmented approaches do eventually outperform the non-augmented approaches in terms of final classification performance in all the experiments. Nevertheless, one can observe that the effect of the augmentation module is dependent on the dataset employed. For the AILARON dataset, with the result shown in figure 6.15e and 6.15f, the performance enhancement is minimal, whereas for the CIFAR dataset, as illustrated in figure 6.15a and 6.15b the difference is significant. A reason for employing the augmentation module is to enhance the performance of the informative metrics relative to the other approaches. The methods believed to benefit from the augmentation module are the DFAL and CIRAL methods, which are dependent on good decision boundaries. Observing from the result shown in figure 6.15b that the DFAL method has a performance increase with the augmentation module employed. This effect can also be observed in figure 6.15g and 6.15h, where the CIRAL method performs best with the augmentation module, and second-worst without.



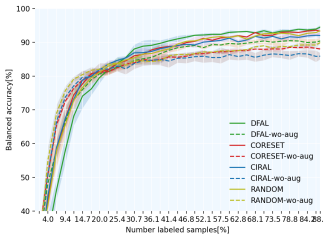
(a) CIFAR.



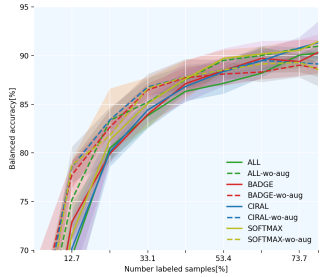
(b) CIFAR.



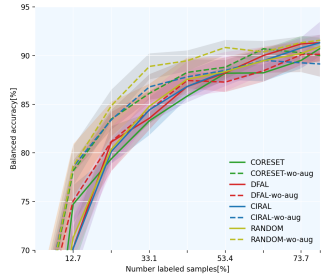
(c) Kaggle.



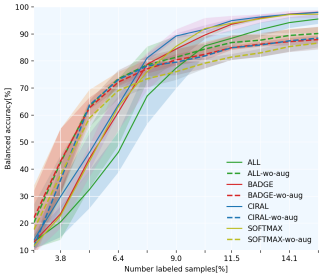
(d) Kaggle.



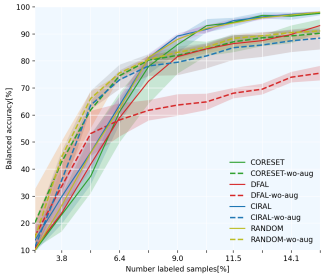
(e) AILARON.



(f) AILARON.



(g) Pastore.



(h) Pastore.

Figure 6.15: Comparison of the effect of data augmentation in AL. The figures on the left-hand side show the comparison of hybrid methods. The figures on the right-hand side show the comparison of informative and representative methods.

Chapter 7

Discussion

This chapter presents a review of the results of the thesis and a discussion on the research questions presented in the introduction. The background material and the conducted experiments presented in chapter 2 and 6, respectively, lay the foundation for the discussion.

7.1 The current standing of research in the field of AL

The different active learning methods presented in the literature are commonly divided into informative and representative approaches, as described in section 2.4. Experimental results from this thesis suggest that many of the informative approaches proposed in the literature suffer from the inability to reliably measure the uncertainty of a deep neural network. This was particularly evident in the experiments presented in chapter 6, where methods relying on the uncertainty of the network, performed worse than random benchmark sampling, RBS. Although uncertainty quantification in the field of AL is paramount, the findings of the literature study conducted in this thesis suggest that few studies have been done on this subject. Several methods for uncertainty quantification (UQ) were presented in [5], however, they reported fewer studies on semisupervised methods compared to the supervised and unsupervised methods. Bayesian neural networks (BNNs) have commonly been applied in the literature to provide information about the reliability of the predictions in deep learning (DL) frameworks [5]. Promising approaches leveraging Bayesian inference frameworks for uncertainty measures [18], more recently representative measures [48] and addi-

tionally meta-learning [37, 22, 31] have been proposed in the AL literature. However, the uncertainty quantification represents a gap in the AL literature that can be filled by further studies to label samples that provide the most information to the classifier.

Transferability and testing in the AL domain

Results from the experiments in chapter 6 show how the performance of different strategies changes with the experimental conditions. Several heuristics for the identification of informative samples have been proposed in the literature and reported to perform better than random benchmark sampling, RBS. However, the experimental results from this thesis suggest that many of the methods proposed in the literature are only performing well under a brittle combination of experimental settings. Concerning this discovery, two aspects of active learning in the literature should be discussed. First, while having received little attention in the evaluation of AL approaches, the transferability property is essential to prove the usability of an AL framework in various domains. The concept of transferability is largely related to the task-agnostic approaches which have gained increasing interest from researchers over the last years [21, 65, 55]. This is an important research direction for making AL robust and applicable for real-world problems. Second, a challenge in the current AL research is the lack of a common ground for testing of the frameworks proposed in the literature. The most common benchmark employed in the literature is RBS, however, as discovered in [43], different papers reports deviating results for RBS under identical settings. This lack of a common testing ground has also led to a disagreement in the literature on the performance of informative and representative methods as indicated in [50]. Consequently, future research should be evaluated on universal benchmark results for standard datasets. Further, another way of mitigating the aforementioned challenges in the AL research would be to create unified datasets where some samples are proven to be more informative or representative than others. In addition to comparing with unified benchmark classification results, proposed AL frameworks could also be measured on the amount of proven informative and representative samples queried.

Methods for efficient plankton taxa labeling in the literature

The literature study conducted in this thesis identified several methods for efficient plankton taxa labeling. However, a minority of the methods proposed in the AL literature have been adopted to the planktonic domain. Instead, clustering based methods and unsupervised learning is prominent in the field [45, 51]. These proposed unsupervised methods do, however, address an important aspect of the plankton taxa labeling, that is, how to identify the number of classes present in an unlabeled dataset. Com-

pared to other AL applications, this challenge is especially prominent in the planktonic domain, as described in section 6.2.3, and should be considered when developing AL frameworks for this application. Furthermore, T-SNE visualizations in chapter 4 show how the plankton classes is highly separated based on their features, suggesting the applicability of representative approaches in the domain. This assumption is further supported by the experimental results presented in section 6.2.4, where the DFAL informative approach is shown to suffer from an unbalanced class querying. As previously mentioned, the literature study conducted in this thesis revealed gaps in the adoption of recent methods from the the AL literature to the planktonic domain. As described in section 3.1.3, the proposed approaches relied on uncertainty sampling based on support vector machines (SVMs) and the softmax layer in CNNs, respectively. The latter has shown to be an unreliable metric for uncertainty in CNNs [17]. However, this gap is addressed in this thesis by adopting novel methods from the AL literature to the planktonic domain. The framework and considerations presented in this thesis are creating a foundation for future adoption of AL in the planktonic domain.

7.2 Considerations and challenges related to AL

Most approaches of AL in the literature consider the situation where a fixed number of samples is queried in each cycle until a budget **B** is exhausted. It is advantageous, yet non-trivial to find the optimal number of samples to query in each AL cycle. The number will, as seen in the experimental results presented in chapter 6, often be dependent on the image classifier, the active learning strategy, and the dataset applied in the framework. As described in section 2.5.1, batch-mode informative approaches are often associated with correlated and redundant samples in the queried batches. A simple, yet effective approach to mitigate this challenge has been to lower the query size. However, as described in section 2.4, fewer data points would lead to a smaller update of the network weights and a higher computational effort from an increased number of AL cycles. This is in particular infeasible for a dense and parameter-rich classifier. However, a low query size proved to be effective in the experiments presented in section 6.2.2, where the overall accuracy dropped with an increasing query size. In addition to redundant sampling, the high query size does also lead to fewer updates of the weights in the image classifier. The queries will, as a result, be from less optimal areas compared with a model trained over more iterations. In general, a lower query size would lead to a more agile classifier in terms of updates on the decision boundary, however, a too low query size would lead to insufficient data presented to the classifier. Moreover, another factor to consider when selecting query size is the applied active learning strategy. In a paper proposed by [44], the author found the core-set method to

outperform random sampling for a high query setting whereas underperforming in a low-budget setting. The core-set approach, which is a representation-reliant method, as described in 2.5.2, needs an initial high amount of images to learn a good representation of the feature space. The results from the paper suggest that the core-set approach and other representation-reliant approaches could benefit from a dynamic query structure. By initially querying a high amount of samples, the representative-reliant approach is enabled to get a better initialization from which it can benefit in the later rounds with lower query size. The choice of query size is dependent on the dataset, the employed image classifier and the AL strategy. In general, a representation-reliant AL strategy would need a high initial amount of labeled samples, whereas other informative-based strategies would necessarily not. Hence, the latter strategy would be more dependent on the image classifier applied and the trade-off between the computational effort and the correlation among the queried samples. Furthermore, in low-budget regimes, a large query size may lead to fewer updates on the classifier, which was observed to be disadvantageous in the experiments presented in section 6.2.2. In addition to the previously described dynamic query structure, another less static query structure could also be applied. By considering the query size as an upper bound rather than a fixed size, one can leave out samples that prove to be less informative or representative with the advantage of having queries left for the coming rounds. This effectively opens the possibility of dynamically making larger queries when the model identifies more valuable samples and lower queries on the contrary.

Strategy- or data-driven implementation

Another consideration in the implementation of AL is whether it should be *data-* or *strategy-driven*, that is, should the employed dataset and domain be considered when selecting an AL strategy. From the experimental results presented in chapter 6, it was largely evident that the performance of the AL approaches differs depending on the experimental conditions. In particular, the results on the Pastore dataset in section 6.2.4 reported that the representative approaches outperformed the informative approaches, whereas the opposite was true on the CIFAR dataset in 6.2.1. Looking behind the results, a large difference in the class separation for the two datasets is observed in figure 4.2 and 4.8, suggesting that the feature separation and data complexity should be considered when selecting a strategy for the AL framework. In other words, the representative approaches are superior on datasets with a large class separation, whereas the informative approaches are performing better on datasets with an intertwined feature space. Furthermore, the experimental conditions in the experiments presented in section 6.2.1 proved to be beneficial for the informative approaches. Looking at the queried class distribution in figure 6.11, it is evident that the

informative approaches, in particular DFAL and ALL, can mitigate the class imbalance present in the Kaggle dataset. As a result, it can be observed in figure 6.9 and 6.10, that these methods outperform the other representative and hybrid methods. This observation suggests that the informative approaches are better at ignoring large and prominent classes compared with the representative approaches. It should be noted that ALL is selecting between an informative and a representative approach in each round, however, in the aforementioned experiment, it was heavily reliant on the informative approach. In general, it can be smart when choosing an AL strategy to get an overview of the feature space of the available data. A challenge with representation-reliant methods is to get a good feature extraction. This can, however, be solved with transfer learning from similar domains as described in section 2.2.3.

Limiting factor of AL

The concept of AL is relying on the idea that some samples bring more information to the classifier than others. However, when this condition is not present, the current strategies developed for AL can not effectively reduce the labeling effort. That is, when all samples are considered equally valuable, the full dataset needs to be labeled to achieve the full accuracy score. This was particularly evident in the experiment on the CIFAR dataset in section 6.2.1. The experimental results showed that the random benchmark sampling (RBS), performed on par or better than the active querying methods. This result suggests that in datasets where much of the training examples are equally valuable, it is difficult to actively query samples better than RBS. The active query methods will instead fail to sample the true data distribution, introducing a bias in the distribution of the labeled samples. However, a dataset representing the opposite was evident for the experiments presented in section 6.2.4, where only a small fraction of the samples was needed to gain full accuracy. In a paper by [61], an investigation was conducted on the information provided in the different datasets. Their findings were aligned with the experimental results in this thesis, showing that for some datasets, the performance of an AL framework is naturally limited by the information distribution.

7.3 Towards a robust framework for the planktonic domain

The framework proposed in this thesis is constructed to address gaps in existing methods in the literature and be a robust method for actively querying samples from the planktonic domain. In particular, the proposed hybridization looks to fully utilize

the relevant knowledge found in the data distribution of the unlabeled data pool while also considering hard examples lying close to the inter-class decision boundaries. By leveraging two independent methods of sampling, the AL becomes more robust in terms of adaptability to different experimental conditions. The experimental results presented in section 6.2 show that the proposed framework is performing well in a variety of experimental conditions. In particular, the framework is handling class imbalance well, as is illustrated in figure 6.10. Further, the results from the experiment conducted on the Pastore dataset illustrated in figure 6.14, suggests that it is an effective strategy also in low-budget regimes. On the contrary, other methods are, as previously discussed, observed to be performing well only for a given combination of experimental conditions. These experimental results suggest that hybridization of AL is beneficial for the robustness and accuracy of the AL strategy.

The effect of data augmentation

To enhance the classification performance, a data augmentation module is employed in the AL framework. From the results in section 6.15b, one can observe how the DFAL approach benefits from having improved decision boundaries and outperforms the other methods. This can also be seen for the proposed CIRAL framework in figure 6.15g and 6.15h. From an initial average performance, it ends up performing on par or better than the rest of the AL approaches. These results indicate the advantage of applying data augmentation and suggest that the improved decision boundary of the classifier is benefiting the informative metric employed. Nevertheless, one aspect to notice here is that the strategies without the augmentation module have an initially better performance on the planktonic datasets. This observation suggests that a larger labeling budget should be employed when applying a data augmentation module to the AL framework. Furthermore, it is observed large differences in the effect of the augmentation module. For complex and information-rich datasets such as the CIFAR, one can observe a large performance gain, as illustrated in figure 6.15a and 6.15b. However, the performance enhancement on the AILARON dataset, as seen in figure 6.15e and 6.15f, is less significant. Nevertheless, both the Plankton and the Kaggle dataset significantly increase their performance with the augmentation module, suggesting its applicability for datasets with high image quality.

Challenges and improvements

While having a robust classification performance, the proposed framework suffers in the later AL cycles of not identifying informative samples equally well as the pure informative sampling. This is prominent in the results of the experiments presented in

section 6.2. Ideally, the hybridization should have the high performance of the pure informative metrics in the later rounds, and the robust performance of the representative metrics in the early rounds. The trade-off function and representative metric employed to avoid redundant sampling and incorporate the full data distribution are considered to be the reasons for the performance degradation in the later rounds. In particular, a pitfall with the representative metric is the querying of samples from sparse, outlier regions that can confuse the model. The framework is constructed to be aware of the latent space, and the performance degradation at the later AL cycles can indicate that informative samples are traded for samples from sparse regions. The framework should be improved to be further applicable for real-world applications. In particular, a smarter trade-off function could be employed to utilize more high informative samples in the later queries, and the representative metric should avoid querying samples from outlier regions as they can be considered as noise in the dataset. The hybrid framework will provide a stable and high accuracy if this challenge can be overcome.

Chapter 8

Conclusion

This thesis assesses the field of active learning as a way of minimizing the manual effort in plankton taxa labeling. The work resulted in a proposed hybrid active learning framework that combines metrics for representative and informative sampling. The framework has been compared to other state-of-the-art AL approaches and has proven to achieve a more consistent classification performance in a variety of experimental conditions.

The literature study conducted throughout this thesis identified two broad categories, informative and representative approaches, that the proposed methods in the AL literature can be divided into. The informative approaches aim at identifying samples the classifier is uncertain about and are often reliant on the uncertainty quantification (UQ) of the employed classifier. These approaches can in theory identify hard examples for labeling, however, current methods for UQ are not reliable, and will often lead to a bias in the queried data distribution. The representative approaches, utilizing the latent space, have shown to be a good way of capturing the data distribution of unlabeled samples and mitigate the aforementioned bias. However, representative methods in the literature have been shown to fail when the dataset includes many hard examples or outliers. Hybrid approaches have been proposed in the literature as a midway, combining the informative and representative approaches, to both utilize the full data distribution and identify hard examples.

In the related work chapter of the thesis, methods for minimizing the manual effort for plankton taxa labeling were presented along with AL methods related to the proposed framework. The chapter presents a gap in the development and adoption of

AL methods to the planktonic domain, which the framework proposed in this thesis intends to mitigate. To identify a representative metric for the hybrid framework, two different clustering-based methods were implemented and tested. The core-set approach achieved high accuracy in the planktonic domain and was employed in the framework to utilize the full data distribution and avoid redundant sampling in the AL queries. To mitigate the challenge with uncertainty quantification in the softmax layer of CNNs, a distance-based approach employing adversarial attacks was implemented. This informative metric is dependent on good decision boundaries to get full utilization, and for this reason, a data augmentation module was employed to enhance the performance of the classifier. The augmentation module allowed the classifier to be more robust on complex data structures that exist in the plankton dataset and improve its performance relative to other active learning strategies.

Several different approaches representing the broad categories in the AL literature were implemented and compared together with the proposed framework to gain an overview of potential challenges related to the implementation of AL. The output of the results showed that informative AL approaches perform best when the dataset is unbalanced or has intertwined classes. Moreover, representative approaches proved to be superior when the dataset has a large class separation. It was further shown that the query size in the AL frameworks largely affected the performance, and it was evident that representative approaches benefit from a large query size whereas a lower query size is advantageous for informative approaches in low-budget regimes. The study on the implementation of AL suggests that the choice of strategy should be driven by the dataset it is applied to. Considerations when picking strategy includes query size, budget size, class separation, and data complexity. Finally, it was shown that random benchmark sampling can outperform active learning when the dataset it is applied to mostly contains valuable examples, in that case, AL has shown to be a less effective way of minimizing the manual effort of labeling.

The proposed framework has shown promising potential as a tool for minimizing the manual effort for plankton taxa labeling. Among the tested approaches, the proposed framework gave the best overall performance on the planktonic datasets, indicating its applicability for the AILARON project. It further proved to handle datasets with class imbalance and situations with low-budget regimes well. The proposed AL framework can work as valuable support for domain experts, and with additional improvements, further reduce the number of manual labels needed to achieve a consistent and satisfactory classification performance on planktonic datasets.

Chapter 9

Future work

Throughout the work with this thesis and the development of the proposed framework, new challenges have been faced, and ideas to solve them have been born. This chapter presents promising research directions that are both in general for the field of AL and also specific for the proposed framework. Hopefully, some of these future research directions would encourage further research on active learning for the planktonic domain.

- **Better way of trading off the informative and representative metric.** A potential improvement on the query selection should improve on the informative sampling at later AL cycles. As evident in section 6.2.3, the proposed framework is performing worse than the sole informative metric in the later AL cycles, indicating that it is not able to efficiently leverage the informative metric in the hybrid framework. A promising future research direction would be to find a trade-off function that would put more weight on the informative samples at the later AL cycles. A part of the solution for this problem could be to implement a dynamic query structure in the AL framework.
- **Find other representative metrics that are not reliant on feature extraction.** The current representative metric employed in the AL framework is heavily reliant on learned representations of the unlabeled data. Consequently, the framework is dependent on leveraging a high amount of initial queried samples along with transfer learning to incorporate initial knowledge into the feature extractor. A promising future research direction would be to identify and implement alternative methods for representative sampling that are not as reliant on a trained feature extractor. A particularly interesting addition to the framework would have been the Bayesian sparse-set approach presented in section 2.5.2.

- **Make the informative metric more task-agnostic.** The informative metric proposed in the thesis is reliant on the decision boundaries and the classification performance of the employed classifier. This makes the hybrid method vulnerable to other experimental conditions where another classifier may be employed. To mitigate this, a promising future research direction would be to make the adversarial attacks, described in section 2.5.1, on a separate network module, independent of the image classifier. A relevant implementation question would be how to train this separate network, however, this question along with the task of making the framework more task-agnostic are left for future work.
- **A way of finding the number of classes present in the unlabeled dataset.** An essential and promising future direction would be to add a module to identify the number of different classes present in the unlabeled dataset. Relevant studies and implementation of similar approaches have been conducted in [51, 45]. This would be largely beneficial when applying the AL module to real-world applications. Evidently in the AILARON dataset described in section 4.3, the 'Other' category could be mitigated by implementing the aforementioned module.
- **Find better ways of measuring the informative value in samples.** As discussed in 7.1, a prominent challenge in the field of AL is to confidently identify samples in which the classifier is uncertain. A promising future research direction would be to investigate methods for uncertainty estimation in deep learning and adopt them to an active learning framework.
- **Compare the proposed framework with task-agnostic approaches.** The proposed AL framework has shown a high level of transferability in the experimental results presented in chapter 6. However, to get an indication on how well the proposed hybrid framework performs in terms of transferability, additional experiments should be conducted comparing the performance with state-of-the-art task-agnostic approaches. Relevant approaches to compare with include the learning-loss and variational autoencoder described in section 2.5.4.
- **Construct classifier models that require a minimum amount of labeled datasets for training and embed those models into AUV platforms for in-situ plankton classification.** This future research direction is broad and concerns the application of an AL framework to real-world applications to minimize the labeling effort for biologists. The proposed framework has proven a high classification performance when adopted to the planktonic domain, however, some adjustments in the implementation are needed to apply it for real-world applications.

Appendix A

Submitted papers

A combined informative and representative active learning approach for plankton taxa labeling

Martin Lund Haug[1], Aya Saad[1], Annette Stahl[1]

[1]Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, NTNU, Trondheim, Norway

ABSTRACT

With an ever-increasing amount of image data, the manual labeling process has become the bottleneck in many machine learning applications. Plankton taxa labeling is especially a challenge due to its complex nature, and the manual labeling effort places a large burden on the domain experts. The Active Learning (AL) paradigm is a promising research direction adopted in the literature to minimize the manual labeling effort exerted by domain experts. Many approaches for AL have been proposed over the recent years to improve the labeling task by supporting the construction of large datasets suitable to train machine learning models while minimizing human involvement in the process. Our empirical study suggests that many modern active learning methods fail to incorporate both the samples that represent the statistical pattern of the data and the samples in which the machine learning model is not confident about.

Inspired by these limitations, we propose an algorithm that combines these two types of sampling in order to capture the data distribution of the whole feature space, prevent redundant sampling from correlated uncertainty queries and fine-tune the inter-class decision boundary. Our experiments show that the proposed method outperforms each of the methods separately. Further, it also proves to be efficient on both the CIFAR dataset and the more complex Kaggle plankton dataset.

Keywords: image analysis, deep learning, plankton taxa distribution, active learning

1. INTRODUCTION

Convolutional Neural Networks (CNN) models have proved competent at solving computer vision problems in the paradigm of the Supervised Machine Learning (ML) approaches. However, to make such models reliable, an immense amount of pre-classified input datasets is required in the training process. Constructing such large datasets requires an extensive manual effort for labeling, which requires a massive amount of time. Nevertheless, the resulting manual classification is imperfect and prone to errors.

Active learning (AL) is a promising research direction of machine learning that aims at mitigating the burden of human experts on labeling training instances. They do so by exploiting the fact that not all samples bring equally much information to an image classifier.[1] Therefore, by finding only the most informative samples and query them for manual labeling, the classifier can be trained to achieve equal performance as if it was trained on the whole dataset.[2] Existing AL models in the literature can be classified based on the unlabeled data readiness and the sampling pool chosen. In other words, when data arrives in streams, the AL model is considered as a stream-based model,[3] while it is pool-based[4] otherwise. Further, the AL models' mode of sampling varies between batch-mode[5] or single-mode[4] depending on the number of data samples presented and chosen at each labeling round. With the recent developments of convolutional neural networks (CNN), batch-mode sampling has become increasingly relevant as it is not computationally feasible to update a large network with single data points.

For the sampling modes aforementioned, there are differences among them in how samples are queried for labeling. Mainly, the most important difference is in choosing between informative and representative samples. While the former aims to find samples which the image classifier finds most informative, the latter exploits the feature space of the data points to best capture the data statistical patterns. There exists a broad literature on active learning. The reader can refer to a survey[6] on active learning based on traditional methods of ML, and more recently, the survey[7] on deep learning versions of AL techniques.

Previous work has shown that active learning has proven to be an effective way of choosing informative samples from a large number of unlabeled samples.[8–10] Although hybrid methods that combine informativeness and representativeness are increasingly popular among researchers, much of the existing methods only incorporate either informativeness or representativeness. Existing AL models lack efficient utilization of the feature space of the dataset under consideration.

They sometimes select samples for the training that fail to fit the different classes' representation specifically when the boundaries between the classes have some overlap. The resulting proposed samples suffer over-fitting or under-fitting the dataset. Therefore they affect the performance of the classifier. Furthermore, in [1], the authors investigated how different datasets had various amounts of information incorporated in the images. The study found that for some datasets,[11] a few representative samples were enough to capture the data statistical distribution. However, for other datasets,[12] this proved not to be true. Thus, the success in employing a stand-alone uncertainty or representative sampling mode is dependent on the dataset.

Therefore in this paper, we propose an efficient algorithm that combines an informative metric with a representative metric approach for active learning. The proposed algorithm begins with a focus on the diverse feature space. It gradually focus more on samples located at the proximity of the classes' decision boundaries to further fine-tune the machine learning model. The aim behind this hybridization has three folds: 1) the novel model will have a good initialization from incorporating the full feature space in the early rounds of query and training. Inspired by the work in [13], a trade-off function gradually moves the focus from diverse samples to more uncertain samples during the training process in order to fine-tune the model with samples located at the boundaries of the classes representations. 2) Adding diversity sampling to the queried uncertainty samples prevents redundant labeling representation from the same area of uncertainty. 3) As the softmax layer on neural networks have shown to be a bad proxy for the uncertainty of neural networks,[7,14] an adversarial active learning method is employed. This method has previously shown good results,[15] however it was not employed with sub-modular heuristics as is done in this paper.

Experiments are conducted on the plankton dataset from National Data Science Bowl[16] and the CIFAR dataset.[12] The ResNet-18[17] architecture is employed as the learning network model. It is worth noting that no data augmentation was performed on neither the CIFAR nor the plankton dataset, as is often done to enhance the performance on classification. The results demonstrate that the proposed algorithmic framework is more efficient compared to each of the strategies separately. Furthermore, they have shown that the novel proposed hybrid algorithm is more effective when dealing with difficult datasets such as the plankton.

The rest of the paper is organized as follows. Section 2 introduces some preliminary knowledge related to this paper. Section 3 presents related work in the area of active learning, and hybrid sampling methods in particular. Section 4 explains our proposed algorithmic framework. Section 5 presents the experimental results. Finally, in section 6, a conclusion is made on the the contributions of this paper and also future directions are presented.

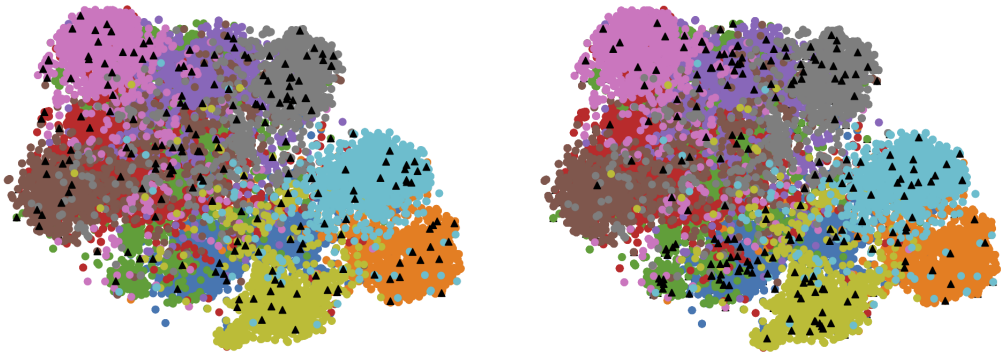
2. BACKGROUND

Active learning is a type of semi-supervised learning that provides classification accuracy competitive with fully-supervised learning approaches, while having the benefits of minimal human interaction from unsupervised learning. The main principle is to iteratively pick subsets from the available unlabeled data in order to build a training set for a machine learning model. As described in the previous section, the query methods of active learning can primarily be categorized into methods that exploit the feature of the data and methods that search for samples the machine learning model finds informative. A way of finding the latter has often been done by finding samples the learning model is uncertain about, e.g. samples in the proximity of the inter-class decision boundaries.

A large number of methods for finding uncertainty samples have been proposed in the recent years due to their simplicity and comprehensiveness. Many of these have been based on the softmax layers of CNNs as a proxy for the networks' uncertainty. Such an approach was proposed by [18], who in addition pseudo-labeled high confidence samples for additional robustness. However, research[7,14] has shown that these softmax probabilities work as a bad proxy for the confidence of neural networks, and will often lead to worse performance than random benchmark sampling. Consequently, other ways of measuring the uncertainty of neural networks have been proposed in the later years. [8] proposed a way of creating an ensemble of network architectures by using Monte Carlo dropout and measure the disagreement in prediction among the networks. A conceptually equal method have also been studied in [9] where the authors employed an ensemble of different CNNs instead of MC dropout. A drawback with these ensemble methods is the computational effort that is increasing with the dimensions of the learning network and number of unlabeled samples.

A different approach from using the classification results of the learning networks has been proposed by [19] to calculate the distance to the inter-class decision boundary. Samples lying close to the decision boundary are considered to be informative for the machine learning model as they can help fine-tuning the model parameters. However, as it is feasible

for support vector machines (SVM), it is a more complex operation for CNNs. Nevertheless, to transfer this approach to CNNs, [15] proposed a way of measuring the distance by making adversarial attacks and find which of the images change the classification. By ranking the size of the perturbation needed to change the sample classification, one can get a proxy on how far the sample is from the decision boundary. This method of looking at the input to the network is somewhat the other way around of looking at the softmax layer as done in [18]. However, both of the latter methods queries the top most uncertain images. As can be seen in figure 1 and also stated in [20], uncertainty sampling tend to lead to high correlation among the samples leading to a lack of utilization of the data distribution and also the labeling of redundant samples. From



(a) Core-set representative metric. (b) DeepFool informative metric.

Figure 1: T-SNE plot of 200 samples queried with a representative metric and an informative metric. The different colored data points represent the images of the 10 different classes from the CIFAR dataset. With the T-SNE algorithm,[21] the images are projected to the two-dimensional feature space.

figure 1 (a) one can observe that by employing a representative metric exploiting the full feature space of the available data points, this problem can be overcome. A large number of methods for finding such representative samples have been investigated over the later years. They can be roughly divided into categories that tries to exploit the feature space and others that aims to maximize some performance metric. An example of the latter is, as proposed in [10], a method that approximates the complete data posterior of model parameters that produces diverse batches. By selecting sub samples, the method tries to lower the expected value of the loss function. An example of the former is, as proposed in [22], a diversity method that performs a farthest-first traversal to cover the feature space. A similar example is shown in [20] that proposes a core set method to find clusters based on the min max facility location problem and then optimize these clusters with mixed integer programming. The overall aim for most of the representativeness methods is to replicate the distribution of the complete unlabeled set. By regarding active learning as a binary classification task between the labeled and unlabeled sets, [23] aims to make the labeled dataset indistinguishable from the unlabeled dataset by capturing the statistical distribution of the unlabeled data.

3. RELATED WORK

This paper proposes a hybrid active learning framework combining the informative and representative sampling strategies described in section 2. This hybridization of active learning has become increasingly popular among researchers in later years. In [24] a method of combining predictive entropy based uncertainty sampling and a distance function on a learned feature space to optimize the selection of unlabeled samples was proposed. Their method was based on the assumption that the most informative samples are the ones where the model has the highest uncertainty and greatest distance to the existing training examples. In [13], the authors aim to fine-tune pre-trained networks with a combination of informative and representative samples. They are also employing a trade-off parameter to let the representative samples have high influence in the beginning, and gradually use more uncertain samples. Further, [25] proposed a hybrid method to deal with the imbalanced classification problem. Their uncertainty metric was based on the probability output from the neural network, while their diversity metric was based on distance from k-means clusters on already labeled data points. Similarly, [26] proposed a work of diversified subset selection that use classical methods of uncertainty, margin sampling and entropy

from the softmax probability distribution. To find diverse samples they used, similar to this paper, min-max facility location in addition to disparity minimum. An important finding in the paper suggested that similar data points within a class made disparity-min pick outliers and thus confuse the model. Moreover, instead of using the output layer probabilities directly, [5] proposed a hybrid method that uses the size of the backward gradient to incorporate the uncertainty metric. For the diversity the authors employed the *k-means++*[27] algorithm. Furthermore, [28] proposed a method to increase diversity in mini-batch active learning. Their experiments showed that diversity-enhancing approaches outperformed a baseline of uncertainty sampling methods. They combined informativeness with representativeness by using margin sampling from the softmax layer as uncertainty metric and the k-means algorithm as representative metric.

The aforementioned related work are often reliant on the output layer of the network employed as an uncertainty metric. Motivated by this, we build on the uncertainty metric proposed in [15] and employ it with a min max sub-modular heuristic to form a hybrid active learning method. Furthermore, similar to [13], we combine the metrics with a trade-off parameter. This is aligned with the findings of [5], who observed that it is advantageous to do representative sampling early in the training then in later rounds focus more on informative samples to fine-tune the model.

4. PROPOSED FRAMEWORK

The framework introduced in this paper combines the informative metric of an adversarial attack with the representative metric of the facility min max problem. Figure 2 illustrates how these methods are combined in the proposed framework. From the plot of the informative metric shown in figure 1 (b), one can observe that the queried samples have high correlation in some areas; this suggests that there exists some redundancy among the queried samples. By incorporating a representative metric to the active learning framework, one can choose the informative samples that also best represent the feature space. Moreover, with a trade-off function initially incorporating all samples, the learning network will gain an overview of the whole feature space. As the training proceeds and general decision boundaries are formed, more focus is put on samples on the inter-class decision boundaries. By switching focus to these samples, the learning model is able to fine-tune the decision boundaries to handle examples that are difficult to classify. As described in algorithm 1, the number

Algorithm 1 CIRAL: Combined informative and representative active learning

Require: Unlabeled samples D_0^U
Require: Initially labeled samples D_0^L
Require: Query budget **B**
Require: Batch size β
Require: Set of hyper-parameters to train the network $\mathcal{H}$
Require: Trade-off constant $\mathbf{K}_0$
Require: Trade-off rate $\delta \in (0, 1)$
  $\mathbf{K}_k = \mathbf{K}_0$
  $D_k^L = D_0^L$
  $D_k^U = D_0^U$
  while $D_k^L - D_0^L \leq$ **B do**
    $\mathcal{A}_k = \textsc{Train}(\mathcal{H}, D_k^L)$
    for $x_i \in D_k^U$ **do**
      $r_i \leftarrow \textsc{DeepFool}(x_i, \mathcal{A}_k)$
    end for
    $b_i \longleftarrow \textsc{TradeOff}(r_i, \mathbf{K}_k)$
    $Q_k \longleftarrow \textsc{MinMax}(b_i, \beta)$
    $D_{k+1}^L \longleftarrow D_k^L \cup Q_k$
    $D_{k+1}^U \longleftarrow D_k^U \backslash Q_k$
    $\mathbf{K}_{k+1} \leftarrow \mathbf{K}_k \cdot \delta$
  end while

of samples going from the informative metric to the representative metric is lowering with a rate δ each round, indicating that the algorithm prioritize samples with high level of informativeness at the later AL cycles. The active learning cycle described is continued until a labeling budget **B** is exhausted.

As illustrated in figure 2, a neural network is trained on the labeled pool in each iteration, forming the decision boundaries used by the informative sampling method. However, as the training proceeds and the model becomes more confident, the decision boundaries become more static, thus it becomes increasingly important to put weight on the samples that are at the proximity of the boundary rather than samples far away from it. This is done by filtering out the samples with the largest distance result from the informative sampling, illustrated with module 5 in figure 2. To find this distance, the informative metric employed uses the DEEPFOOL[29] algorithm to compute adversarial attacks in order to find a proxy for the distance to the decision boundary. The DEEPFOOL algorithm finds the closest hyperplane for each sample and then pushes the sample beyond it with a minimal possible perturbation.

Moreover, to find the representative samples in the next step, the min max facility location problem, well known from literature and described in [30], is employed. It can be formally described as

$$\min_{s^1:s^1\leq b} \max_{i} \min_{j\in s^1\cup s^0} \triangle(x_i, x_j) \tag{1}$$

Where $\Delta(x_i, x_j)$ represents the Euclidean distance between the data points x_i and x_j. Further, s^1 and s^0 is the new queried samples and existing pool of samples, respectively. The optimization problem in 1 can be understood as choosing *b* cluster centers such that the largest distance from any single point to its nearest cluster center is minimized. As this problem is NP-hard, a sub-optimal solution is found by a greedy algorithmic approach as described in [20]. This method is proven to have a solution such that

$$\max_{i} \min_{j\in s^1\cup s^0} \triangle(x_i, x_j) \leq 2\ X\ OPT \tag{2}$$

is satisfied, where OPT is the optimal solution to the optimization problem in 1.[30] As described in our framework, the representative and informative metrics are combined through a trade-off function that gradually puts more focus on the informative samples at the cost of the representative samples. As the DEEPFOOL algorithm returns a list of samples based on their distance to the decision boundary, the trade-off function is only passing on a fraction $\mathbf{K}_k$ of the most informative samples to the representative function. Thus, the algorithm will eventually ignore samples found at large distances away from the decision boundary. Formally, this trade-off method can be described as

$$Q_k = \text{MINMAX}(\mathbf{K}_k \cdot \text{DEEPFOOL}(\mathbf{X}), \beta) \tag{3}$$

Where $\mathbf{K}_k$ is the trade-off constant, $\mathbf{X}$ is the input from the unlabeled samples and β is the number of samples to be queried.

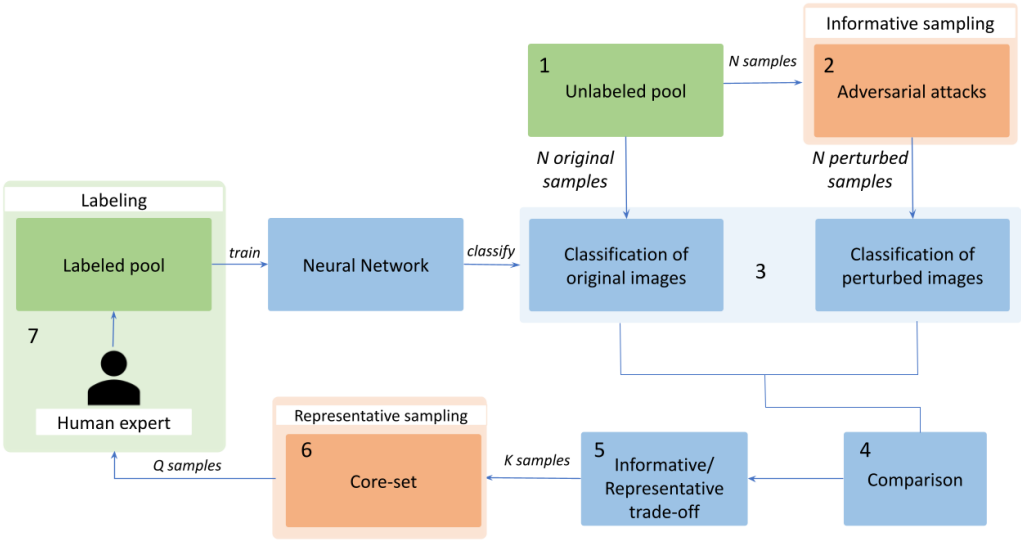


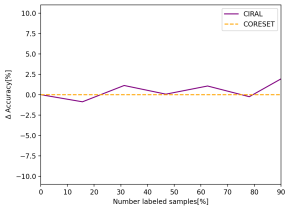
Figure 2: Block diagram of the active learning framework showing the relevant modules and the flow of samples from the unlabeled pool to the labeled pool.

5. EXPERIMENTAL RESULTS

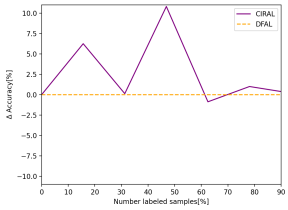
The experiments were performed on subsets of the CIFAR[12] and Kaggle National Data Science Bowl[16] (plankton) datasets, both containing 10 different classes. After each round of querying, the neural network got trained on the labeled pool until convergence of validation accuracy. A prediction was then performed on a separate testing set.

In figure 3 (a-c), results from the proposed hybrid method tested on the plankton dataset are presented. Further, in figure 3 (d-f), the results on the CIFAR dataset are presented. For both datasets, the result for the hybrid method is plotted relatively to the accuracy of the DFAL,[15] core-set[20] and random strategy. Results are also presented in figure 3 (g-h) for the CIFAR and plankton datasets, respectively. One can observe that the hybridization is performing steadily better than the other methods separately. As can be seen in figure 3 (a-c), the proposed method gains advantage on combining representative and informative metrics on the more complex plankton dataset. Looking at figure 3 (a), the method is performing better than the core-set method because it is better at choosing samples that fine-tune the decision boundaries. Further, looking at figure 3 (b) the hybrid method is clearly better in the early rounds of training suggesting that the incorporated representative samples in the early rounds are beneficial for the model.

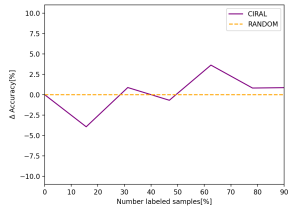
Looking at the graphs in figure 3 (g-h), the proposed method is consistently having high accuracy compared to the other methods. Observing from the results in table 1 and 2 that the proposed method is achieving good results overall, it is especially prominent on the CIFAR dataset.



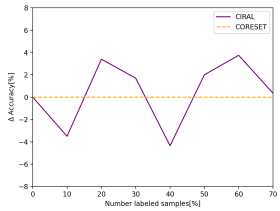
(a) Comparison of CIRAL and a representative method.



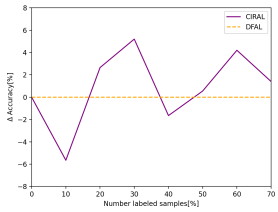
(b) Comparison of CIRAL and an informative method.



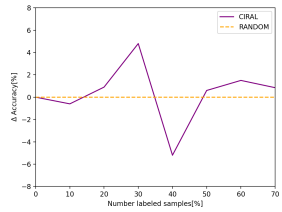
(c) Comparison of CIRAL and the random benchmark.



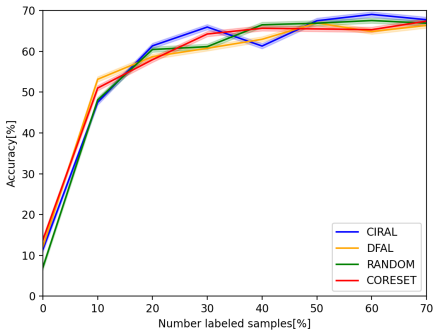
(d) Comparison of CIRAL and a representative method.



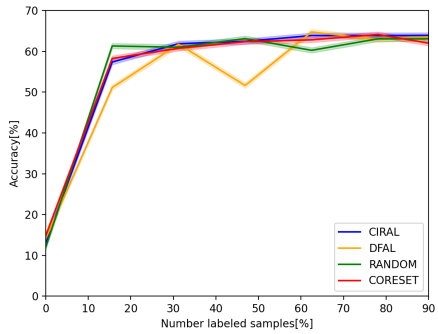
(e) Comparison of CIRAL and an informative method.



(f) Comparison of CIRAL and the random benchmark.



(g) Comparison of CIRAL and other hybrid methods.



(h) Comparison of CIRAL and other informative/representative methods.

Figure 3: Results from experiments with our proposed framework (CIRAL). Results in plot (a-c, g) are from experiments on the plankton dataset. Results in plot (d-f, h) are from experiments on the CIFAR dataset.

Method/Round	1	2	3	4	5	6	7
CIRAL	0.1	0.574	**0.62**	0.624	0.64	**0.64**	**0.64**
DFAL	0.1	0.511	0.62	0.52	**0.647**	0.63	0.638
CORE-SET	0.1	0.573	0.61	0.62	0.63	0.64	0.61
RANDOM	0.1	**0.61**	0.61	**0.63**	0.60	0.63	0.62

Table 1: Classification accuracy from the experiments with our proposed framework on the plankton dataset.

Method/Round	1	2	3	4	5	6	7	8
CIRAL	0.1	0.48	**0.614**	**0.66**	0.613	**0.68**	**0.69**	**0.68**
DFAL	0.1	**0.532**	0.59	0.608	0.63	0.67	0.65	0.66
CORE-SET	0.1	0.51	0.58	0.64	0.65	0.655	0.653	0.67
RANDOM	0.1	0.48	0.605	0.612	**0.66**	0.67	0.676	0.67

Table 2: Classification accuracy from the experiments with our proposed framework on the CIFAR dataset.

6. CONCLUSION

This paper presents a new framework furthering the field planktonic image analysis. Manual labeling of planktonic data is time consuming and puts a large burden on the domain experts. The proposed active learning method is able to minimize this effort while still achieving satisfactory classification results. The framework presented in this paper combines metrics for representative and informative sampling and achieve better performance than each of them separately. The method has proven to be efficient on both the benchmark CIFAR dataset and the more complex plankton dataset, suggesting that these metrics should be considered in combination when applying active learning. The informative metric employed in the proposed framework is dependent on good decision boundaries to get full utilization. An interesting future direction would therefore be to investigate how other representative functions affect the performance of the classifier. In particular, looking at combining Bayesian-based representative metrics with the informative metrics employed in this framework is an interesting direction.

ACKNOWLEDGMENTS

This research is funded by the Research Council of Norway (RCN) IKTPLUSS program (project number 262741) and supported by NTNU AMOS (RCN project number 223254).

REFERENCES

[1] K. Vodrahalli, K. Li, and J. Malik, "Are all training examples created equal? an empirical study," CoRR **abs/1811.12569** (2018).

[2] K. Chitta, J. M. Alvarez, E. Haussmann, and C. Farabet, "Less is more: An exploration of data redundancy with active dataset subsampling," CoRR **abs/1905.12737** (2019).

[3] V. Krishnamurthy, "Algorithms for optimal scheduling and management of hidden markov model sensors," IEEE Transactions on Signal Processing **50**(6), 1382–1397 (2002).

[4] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," CoRR **abs/cmp-lg/9407020** (1994).

[5] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal, "Deep batch active learning by diverse, uncertain gradient lower bounds," CoRR **abs/1906.03671** (2019).

[6] B. Settles, "Active learning literature survey," (2009).

[7] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, X. Chen, and X. Wang, "A survey of deep active learning," (2020).

[8] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," CoRR **abs/1703.02910** (2017).

[9] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler, "The power of ensembles for active learning in image classification," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9368–9377 (2018).

[10] R. Pinsler, J. Gordon, E. Nalisnick, and J. M. Hernández-Lobato, "Bayesian batch active learning as sparse subset approximation," (2021).

[11] Y. LeCun and C. Cortes, "MNIST handwritten digit database," (2010).

[12] A. Krizhevsky, "Learning multiple layers of features from tiny images," (2009).

[13] S. Huang, J. Zhao, and Z. Liu, "Cost-effective training of deep cnns with active model adaptation," CoRR **abs/1802.05394** (2018).

[14] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," CoRR **abs/1706.04599** (2017).

[15] M. Ducoffe and F. Precioso, "Adversarial active learning for deep networks: a margin based approach," CoRR **abs/1802.09841** (2018).

[16] "Plankton imagery data collected from f.g. walton smith in straits of florida from 2014-06-03 to 2014-06-06 and used in the 2015 national data science bowl (nodc accession 0127422)," (2015). Access: 2020-16-12.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," CoRR **abs/1512.03385** (2015).

[18] K. Wang, D. Zhang, Y. Li, R. Zhang, and L. Lin, "Cost-effective active learning for deep image classification," CoRR **abs/1701.03551** (2017).

[19] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," Journal of machine learning research **2**(Nov), 45–66 (2001).

[20] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," (2018).

[21] L. van der Maaten and G. Hinton, "Viualizing data using t-sne," Journal of Machine Learning Research **9**, 2579–2605 (11 2008).

[22] Y. Geifman and R. El-Yaniv, "Deep active learning over the long tail," CoRR **abs/1711.00941** (2017).

[23] D. Gissin and S. Shalev-Shwartz, "Discriminative active learning," CoRR **abs/1907.06347** (2019).

[24] A. Smailagic, P. Costa, H. Young Noh, D. Walawalkar, K. Khandelwal, A. Galdran, M. Mirshekari, J. Fagert, S. Xu, P. Zhang, and A. Campilho, "Medal: Accurate and robust deep active learning for medical image analysis," in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 481–488 (2018).

[25] H. Dong, B. Zhu, and J. Zhang, "A cost-sensitive active learning for imbalance data with uncertainty and diversity combination," 218–224 (02 2020).

[26] V. Kaushal, A. Sahoo, K. Doctor, N. R. Uppalapati, S. Shetty, P. Singh, R. K. Iyer, and G. Ramakrishnan, "Learning from less data: Diversified subset selection and active learning in image classification tasks," CoRR **abs/1805.11191** (2018).

[27] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," tech. rep., Stanford (2006).

[28] F. Zhdanov, "Diverse mini-batch active learning," CoRR **abs/1901.05954** (2019).

[29] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2574–2582 (2016).

[30] D. S. Hochbaum and D. B. Shmoys, "A best possible heuristic for the k-center problem," Mathematics of operations research **10**(2), 180–184 (1985).

CIRAL: a hybrid active learning framework for plankon taxa labeling

Martin Lund Haug ** **Aya Saad** *** **Annette Stahl** ****

Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, NTNU, Trondheim, Norway
** *(e-mail: martlh@stud.ntnu.no)*
*** *(e-mail: aya.saad@ntnu.no)*
**** *(e-mail: annette.stahl@ntnu.no)*

Abstract: With the complex structure of planktonic species and an immense amount of data captured from autonomous underwater vehicles (AUVs), a large burden is placed on the domain experts for plankton taxa labeling. At the same time, the most prominent machine learning (ML) methods for classification rely heavily on a massive amount of labeled datasets to create and train neural network classifier models that perform their tasks accurately. Active Learning (AL) is a ML paradigm that reduces this manual effort by proposing algorithms that support the construction of the training datasets, thus enlarging the sets while minimizing human involvement. To build the training set, AL methods apply heuristics to select a subset of images, i.e., samples, from the entire data. The selected samples that capture the common statistical patterns or feature space are likely to include all the information needed for the training and the learning processes. In addition, the algorithm should prioritize samples that are likely belonging to multiple classes, i.e., having close inter-class boundaries, and might lead to model confusion. Many of the current AL approaches fail to incorporate both types of samples representing the statistical pattern and the samples in which the particular machine learning model is uncertain about.
In this paper, we extend our framework which addresses these challenges with an augmentation module to increase robustness of the model and ensure its adaptability to the planktonic domain. We compare the framework with existing hybrid AL techniques and test an adaption of our extended framework on the planktonic domain. The empirical results from the experiments exerted in this paper confirm higher accuracy achieved by the new extended framework.

Keywords: image analysis, deep learning, plankton taxa distribution, active learning, computer vision

1. INTRODUCTION

Planktonic species are critically important to the oceanic ecological structure as they are the basis of the aquatic food web. Hence, by studying temporal variations in plankton taxa distributions, one can achieve a proxy for the development of the oceanic ecosystems.

Progress in the development of autonomous underwater vehicles (AUV) and robotic visual sensing enables the possibility of capturing large amounts of planktonic image data. Further, Convolutional Neural Network (CNN) models have proved competent at solving computer vision problems in the supervised Machine Learning (ML) paradigm. Embedding CNN models into AUV enables identification of plankton taxa distributions in-situ. However, modern CNNs require an immense amount of pre-classified labeled input in order to achieve satisfactory classification performance. Since plankton biomass appears in many different species, forms, and stages depending on the geographical environment and season, pre-classified training data has to be constructed for each different geographical environment, season and image-acquiring system. Consequently, much effort is needed in the manual plankton taxa labeling with a constrained budget that requires domain expertise, i.e., biologists, to identify the complex structure of planktonic organisms.

Active Learning (AL) is a semi-supervised machine learning approach that aims at mitigating this burden placed on the domain experts. The key idea of AL is to capture the data distribution of the full dataset with only a fraction of the samples. This is possible from the fact that not all images bring equal amounts of information to the image classifier (Vodrahalli et al., 2018).

Existing AL models in the literature can be classified based on the unlabeled data readiness and the sampling pool chosen. In other words, when data arrives in streams, the AL model is considered as a stream-based model, (Krishnamurthy, 2002), while it is pool-based otherwise (Lewis and Gale, 1994). Further, the AL models' mode of sampling varies between batch-mode (Ash et al., 2019) or single-mode (Lewis and Gale, 1994) depending on the number of data samples presented and chosen at each labeling round. With the recent developments of CNNs, batch-mode sampling has become increasingly relevant as

it is not computationally feasible to update a large network with single data points.

The most important distinction between the different sampling modes aforementioned is in their prioritization between informative and representative samples. While the former aims to prioritize samples that are at the proximity of the inter-class decision boundaries, the latter exploits the feature space of the data points to best capture the statistical patterns of the data. There exists a broad literature on Active Learning. The reader can refer to the survey presented in (Settles, 2009), and more recently, the survey on deep learning version of AL techniques is elaborated in (Ren et al., 2020).

The promise of removing the bottleneck of manual labeling in machine learning pipelines in addition to progress in the development of deep learning models has brought a surge in AL research. AL has been proven to be an efficient method of querying informative samples from an unlabeled pool of data points (Gal et al., 2017; Yoo and Kweon, 2019). Further, other approaches focusing on exploiting the latent-space structure of unlabeled samples have also been successfully proposed (Sener and Savarese, 2018). Furthermore, hybrid methods combining the informative and representative metric have become increasingly popular among researchers over the later years (Hsu and Lin, 2015). Still, much of the existing AL methods lack efficient utilization of the latent-space structure and often suffer from high correlation among queried samples. Moreover, by only incorporating model-based query methods, many existing AL approaches lack transferability to other deep learning models. In (Vodrahalli et al., 2018), the authors investigated how different datasets had unequal amounts of information distributed among the images. In some cases a few samples were enough to represent the full distribution of the dataset yet in other cases this proved not to be true. The success of active learning often depends on the information distribution of the dataset; hence, it is rarely possible to rely on either representative or informative sampling.

To address this issue, we proposed in Haug et al. (2021) a combined representative and informative active learning (CIRAL) approach that incorporates the full feature space in the early cycles of querying and puts more weight on samples at the proximity of the inter-class decision boundaries at the later cycles. We compared the novel hybrid framework with informative and representative approaches. We proved that this hybridization outperforms the classical AL approaches under the two categories in terms of the overall model accuracy on the CIFAR dataset with minimal possible data presented to the model. The CIFAR dataset was the most utilized in the literature as a benchmark for performance comparison and as a proof of concept.

The aim behind the proposed hybridization has three folds: 1) the model will have a good initialization from incorporating the full feature space in the early rounds of querying and training. 2) Adding diversity sampling to the queried uncertainty samples prevents redundant labeling representation from the same area of uncertainty. 3) As the soft max layer on neural networks have shown to be a bad proxy for the uncertainty of neural networks (Guo et al., 2017; Ren et al., 2020), an adversarial active learning method is employed. This method has previously shown good results (Ducoffe and Precioso, 2018), however it was not employed with sub-modular heuristics as is done in this work.

The contributions in this paper are two folds:

- First, we compare the performance of the novel hybrid framework with other well-known hybrid methods and show that it achieves better accuracy.
- Second, we extend the originally proposed framework with a data augmentation module to increase the robustness of the model, and to ensure the adaptability of the proposed semi-supervised method to the plankton domain with the goal to minimize the burden on domain experts.

The experiments in this paper are conducted on subsets of the plankton dataset from National Data Science Bowl (kag, 2015) and the CIFAR dataset (Krizhevsky, 2009). The ResNet-18 architecture is employed as the learning network model (He et al., 2015) for the CIFAR, whereas a custom network is made for the plankton dataset.We further created a pre-processing module to adapt the images to the deep learning models employed in this paper and speed up the convergence of the training process. Pre-processing operations include normalization of pixel values and resizing of input images to a fixed dimension. Further, as opposed to many other AL studies (Mittal et al., 2019), we employ regularization techniques in order to enhance the classification performance of the AL models and improve their robustness. More specifically, we employ a random horizontal and vertical flip and a random affine transformation.

The rest of the paper is organized as follows. Section 2 introduces some preliminary knowledge related to this paper. Section 3 presents related work in the area of AL, emphasizing hybrid and plankton specific AL methods in particular. Section 4 explains our proposed algorithmic framework. Section 5 presents the experimental results. In Section 6, a conclusion is made on the the contributions of this paper and also future directions are presented.

2. BACKGROUND

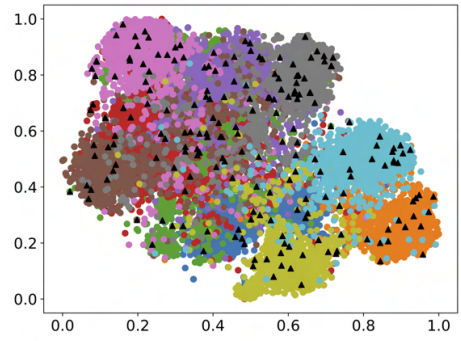
Active Learning is a type of semi-supervised learning that provides classification accuracy competitive with fully-supervised learning approaches, while having the benefits of minimal human interaction from unsupervised learning. The main principle is to iteratively pick subsets from the available unlabeled data in order to build a training set for a machine learning model. As described in the previous section, the query methods of active learning can be primarily categorized into methods that exploit the feature of the data and methods that search for samples the machine learning model finds informative. A way of finding the latter has often been done by prioritizing samples the learning model is uncertain about, e.g. samples in the proximity of the inter-class decision boundaries.

A large number of methods for finding uncertainty samples have been proposed in the recent years due to their simplicity and comprehensiveness. Many of these have been based on the softmax layers of CNNs as a proxy for the

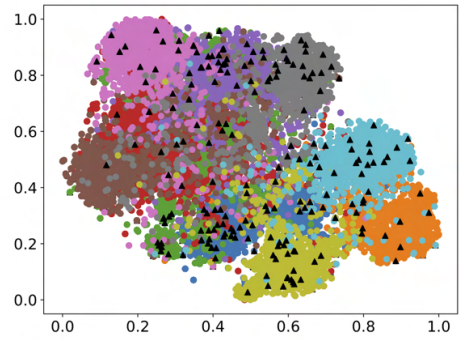
networks' uncertainty. Such an approach was proposed by Wang et al. (2017), who in addition pseudo-labeled high confidence samples for additional robustness. However, research has shown that these softmax probabilities work as a bad proxy for the confidence of neural networks (Guo et al., 2017; Ren et al., 2020), and will often lead to worse performance than random benchmark sampling. Consequently, other ways of measuring the uncertainty of neural networks have been proposed in the later years. Gal et al. (2017) proposed a way of creating an ensemble of network architectures by using Monte Carlo dropout and measure the disagreement in prediction among the networks. A conceptually equal method have also been studied in (Beluch et al., 2018), where the authors employed an ensemble of different CNNs instead of the Monte Carlo dropout. A drawback with these ensemble methods is the computational effort that is increasing with the dimensions of the learning network and number of unlabeled samples.

A different approach from using the classification results of the learning networks has been proposed by Tong and Koller (2001) to calculate the distance to the inter-class decision boundary. Samples lying close to the decision boundary are considered to be informative for the machine learning model as they can help fine-tuning the model parameters. However, as it is feasible for support vector machines (SVM), it is a more complex operation for CNNs. Nevertheless, to transfer this approach to CNNs, Ducoffe and Precioso (2018) proposed a way of measuring the distance by making adversarial attacks and find which of the images change the classification. By ranking the size of the perturbation needed to change the sample classification, one can get a proxy on how far a given sample is from the decision boundary. This method looks at the input to the network rather than the soft-max layer as done in (Wang et al., 2017). However, both of the latter methods queries the top most uncertain images. As can be seen in figure 1 (b) and also stated in (Sener and Savarese, 2018), uncertainty sampling tend to lead to high correlation among the samples leading to a lack of utilization of the data distribution and also the labeling of redundant samples. The experiments in this paper are exerted on the plankton dataset with a custom neural network.

From figure 1 (a) one can observe that by employing a representative metric to exploit the full feature space of the available data points, this problem can be overcome. A large number of methods for finding such representative samples have been investigated over the later years. They can be roughly divided into categories that try to exploit the feature space and others that aim to maximize some performance metric. An example of the latter is, as proposed in (Pinsler et al., 2021), a method that approximates the complete data posterior of model parameters that produces diverse batches. By selecting sub samples, the method tries to lower the expected value of the loss function. An example of the former is, as proposed in (Geifman and El-Yaniv, 2017), a diversity method that performs a farthest-first traversal to cover the feature space. A similar example is shown in (Sener and Savarese, 2018) proposing a core set method to find clusters based on the min max facility location problem and then optimizing these clusters with mixed integer programming. The overall aim for most of the representativeness methods is to replicate the distribution of the complete unlabeled set. By regarding Active Learning as a binary classification task between the labeled and unlabeled sets, Gissin and Shalev-Shwartz (2019) aims at making the labeled dataset indistinguishable from the unlabeled dataset by capturing the statistical distribution of the unlabeled data.



(a) Proposed points resulting from the representative metric



(b) Proposed points resulting from the informative metric

Fig. 1. T-SNE plot of 200 samples queried with a representative metric and an informative metric. The different colored data points represent the images of the 10 different classes from the CIFAR dataset. With the T-SNE algorithm (van der Maaten and Hinton, 2008), the images are projected onto the two-dimensional feature space.

3. RELATED WORK

Two areas of active learning are related to our work. Firstly, other methods of hybrid active learning have been increasingly popular among researchers in later years. Kaushal et al. (2018) proposed a work of diversified subset selection that utilize methods of least confidence, smallest margin and highest entropy from the softmax probability distribution to find informative samples. To incorporate representative samples they used, similar to this work, min-max facility location in addition to disparity minimum. A similar approach was proposed by Zhdanov (2019)

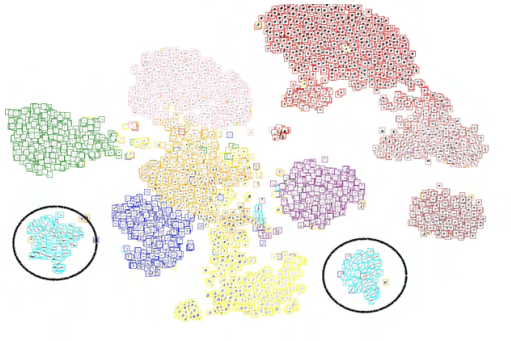


Fig. 2. Visualization of the plankton classes show how the Chaetognath Sagitta class is separated into two groups based on its orientation. The plankton images are project onto the two-dimensional feature space using the T-SNE algorithm (van der Maaten and Hinton, 2008)

to increase diversity in mini-batch Active Learning. Their experiments reported that diversity-enhancing approaches outperformed a baseline of uncertainty sampling methods. They combined informative sampling with representative sampling by using smallest margin sampling from the softmax layer as uncertainty metric and the k-means algorithm as representative metric. In (Huang et al., 2018), the authors aim to fine-tune pre-trained networks with a combination of informative and representative samples. Further, by employing a trade-off parameter, they can let the representative samples have high influence in the beginning, and gradually put more weight on informative samples.

Moreover, instead of using the output layer probabilities directly, Ash et al. (2019) computed a gradient of the predicted category with respect to the parameters of the last layer in the network. To measure the uncertainty of the model, they used this gradients magnitude. Further, to find diverse samples, they collected a batch of samples with the *k-means++* algorithm (Arthur and Vassilvitskii, 2006) to find gradients that span a diverse set of directions. Furthermore, another way of combining informative and representative sampling was proposed by Hsu and Lin (2015). Their method, inspired by the multi-armed bandit problem, would for each iteration explore the performance of different sampling methods and exploit the one with the best performance.

Another field of related work is the plankton specific active learning. Luo et al. (2005) proposed an AL method using multi-class support vector machines (SVM). They used least confidence sampling and margin sampling based on the SVMs decision function to decide which samples to query. Following the developments of CNNs, Bochinski et al. (2018) proposed a deep active learning approach using the probability distribution from the softmax layer as uncertainty metric for the learning model. Equal to Luo et al. (2005), they proposed least confidence sampling and smallest margin sampling in addition to entropy sampling. Additionally, by pseudo-labeling high-confidence they increased the robustness of their learning model, however at the risk of training on erroneous labeled samples. Another approach for minimizing human labeling effort in plankton taxa labeling was proposed by Pastore et al. (2020). Their method utilized fuzzy k-means clustering on extracted features, and a supervised model trained using the k-means clustering labels. Further, they also employed an SVM to do anomaly detection and detect unseen species of plankton.

The aforementioned related work on hybrid AL are often reliant on the output layer probability distribution to work as an uncertainty metric. Additionally, a majority of the proposed hybrid approaches makes no use of modern data augmentation, making it difficult to assess their validity on real applications. Motivated by this, we employ in this paper a data augmentation module as an extension to our original work in Haug et al. (2021) and assess the frameworks applicability to the plankton domain. Further, we compare the results of the novel framework with other well-known hybrid AL methods on both datasets the CIFAR and the plankton datasets.

4. PROPOSED FRAMEWORK

The framework introduced in this work builds on the active learning hybridization proposed in Haug et al. (2021). Figure 3 illustrates how the informative and a representative metric are combined. A data augmentation is added to this framework to increase the robustness of the model and enhance the performance of the informative metric. The reason behind extending the framework with this module is that captured planktonic species have complex structures compared to other datasets; moreover, we found that planktonic organisms from the same class but captured with different orientations are usually split by the models into separate groups as shown by the visualization tool in figure 2.

The data augmentation module, illustrated by module 8 in figure 3, consists of two steps. The first step is the flipping function which randomly generates images horizontally or vertically flipped with 50% probability. The flipping function allows the model to be more invariant to 90°image rotation; The second step is an affine transformation function that is applied with a rotation angle of 7° and with a horizontal and vertical translation of 0.1. This step is used to keep the images center-invariant, thus making the dataset dynamic rather than static which is particularly beneficial for tasks with small amounts of labeled data where overfitting is an issue. This set of augmentation techniques are summarized as $\mathcal{T}$ in Algorithm 1.

Figure 1 shows that the batch of samples queried with an informative metric have high correlation in some areas; this suggests that there exists some redundancy among the queried samples. Based on this inefficiency in sample querying, a representative metric is integrated to the active learning framework. This hybridization enables the algorithm to choose the informative samples that also best represent the feature space of the unlabeled data. Moreover, with a trade-off function initially incorporating all samples, the learning network will gain an overview of the whole feature space. As the training proceeds and general decision boundaries are formed, more focus is put on samples on the inter-class decision boundaries. By switching focus to these samples, the learning model is able

to fine-tune the decision boundaries to handle examples that are difficult to classify.

Algorithm 1 CIRAL: Combined informative and representative active learning extended with the augmentation module

Require: Unlabeled samples D_0^U
Require: Initially labeled samples D_0^L
Require: Query budget **B**
Require: Batch size β
Require: Set of hyper-parameters to train the network $\mathcal{H}$
Require: Set of data augmentation techniques $\mathcal{T}$
Require: Trade-off constant K_0
Require: Trade-off rate $\delta \in (0, 1)$
  $K_k = K_0$
  $D_k^L = D_0^L$
  $D_k^U = D_0^U$
  while $D_k^L - D_0^L \leq \mathbf{B}$ **do**
    $\mathcal{A}_k = \textsc{Train}(D_k^L, \mathcal{H}, \mathcal{T})$
    for $x_i \in D_k^U$ **do**
      $r_i \leftarrow \textsc{DeepFool}(x_i, \mathcal{A}_k)$
    end for
    $b_i \longleftarrow \textsc{TradeOff}(r_i, K_k)$
    $Q_k \longleftarrow \textsc{MinMax}(b_i, \beta)$
    $D_{k+1}^L \longleftarrow D_k^L \cup Q_k$
    $D_{k+1}^U \longleftarrow D_k^U \backslash Q_k$
    $K_{k+1} \leftarrow K_k \cdot \delta$
  end while

As described in algorithm 1, the number of samples going from the informative metric to the representative metric is lowering with a rate δ each round, indicating that more of the informative samples are chosen in the end of the training. After the representative sampling, a number Q_k of samples are queried to a human expert for labeling. This active learning process continues until a labeling budget **B** is exhausted.

As illustrated in figure 3, a neural network is trained on an augmented labeled pool in each round. For the CIFAR dataset, the ResNet-18 architecture is employed as the learning network model. However, for the plankton dataset, a custom network architecture consisting of 3 convolutional layers, 2 max pooling layers and 2 fully connected layers is employed to avoid overfitting and increase generalization.

By increasing the labeled pool with queried samples and updating the parameters of the neural network at each iteration, the inter-class decision boundaries are changing for each round. However, as the training proceeds and the model becomes more confident, the decision boundaries become more static, thus it is becoming increasingly important to put weight on the samples that are in the proximity of the boundary rather than samples far away from it. This is done by filtering out the samples with the largest distance result from the informative sampling, illustrated with module 5 in figure 3. To find this distance, the informative metric employed uses the Deep-Fool (Moosavi-Dezfooli et al., 2016) algorithm to compute adversarial attacks in order to find a proxy for the distance to the decision boundary. The Deep-Fool algorithm finds the closest hyperplane for each sample and then pushes the sample beyond it with a minimal possible perturbation. By adding the aforementioned data augmentation module to the framework, the network will improve its decision boundaries from training on more samples, and resultingly improve the accuracy of the boundary distance proxy provided by the informative metric.

Moreover, to find the representative samples in the next step, the min max facility location problem, well known from literature and described in (Hochbaum and Shmoys, 1985), is employed. It can be formally described as

$$\min_{s^1 : s^1 \leq b} \max_i \min_{j \in s^1 \cup s^0} \triangle(x_i, x_j) \tag{1}$$

Where $\Delta(x_i, x_j)$ represents the Euclidean distance between the data points x_i and x_j. Further, s^1 and s^0 is the pool of labeled and unlabeled data points, respectively. The optimization problem in 1 can be understood as choosing b cluster centers such that the largest distance from any single point to its nearest cluster center is minimized. As this problem is NP-hard, a sub-optimal solution is found by a greedy algorithmic approach as described in (Sener and Savarese, 2018). This method is proven to have a solution such that

$$\max_i \min_{j \in s^1 \cup s^0} \triangle(x_i, x_j) \leq 2\, X\, OPT \tag{2}$$

is satisfied, where OPT is the optimal solution to the optimization problem in 1 (Hochbaum and Shmoys, 1985). As described in our framework, the representative and informative metrics are combined through a trade-off function that only pass on the top K_k samples closest to the decision boundary. Thus, the algorithm will eventually ignore samples found at large distances away from the decision boundary. Formally, this trade-off method can be described as

$$Q_k = \textsc{MinMax}(K_k \cdot \textsc{DeepFool}(\mathbf{X})) \tag{3}$$

Where K_k is the trade-off constant and $\mathbf{X}$ is the input from the unlabeled samples.

5. EXPERIMENTAL RESULTS

The experiments were performed on the CIFAR dataset (Krizhevsky, 2009) and a subset of the plankton dataset of the Kaggle national data science bowl (kag, 2015), both containing 10 different classes. After each round of querying, the neural network got trained on the labeled pool until convergence of accuracy on a held out validation set. A prediction was then performed on a separate testing set. We repeated this process until a pre-defined labeling budget was exhausted. All our results report the average of 3 complete trials. In figure 7, results from the proposed hybrid method tested on the plankton dataset is presented. In figure 7 (a), the accuracy of our method is compared to other hybrid methods. Further, in figure 7 (b), our method is compared to informative and representative methods. Random benchmark sampling is included in both (a) and (b) for reference. One can observe from these results that our proposed method(CIRAL) is performing steadily in terms of classification accuracy and is outperforming the random sampling benchmark by a large margin. The random sampling need approximately twice as many samples to reach the same level of accuracy as our proposed

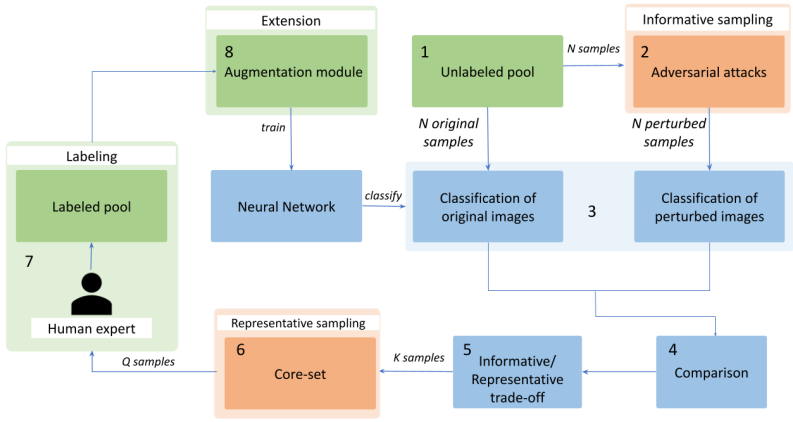


Fig. 3. Block diagram of the Active Learning framework. The process is initiated with an unlabeled pool of N images. An adversarial attack is performed on the unlabeled instances and they are sorted by how much perturbation is needed in order for the neural network to change their classification. This adversarial attack works as a proxy on how far each sample is from the decision boundary, and is an uncertainty metric for the model. Based on the trade-off function, a set of K uncertainty samples are sent to the representative sampling method. Lastly, Q samples with combined informative and representative value are queried to a human expert for manual labeling.

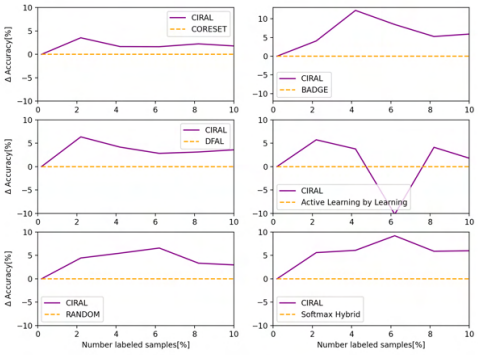


Fig. 4. (LHS) The proposed AL method compared to informative, representative and random methods. (RHS) The poposed AL method compared to other hybrid methods (BADGE, Active Learning by Learning, Softmax Hybrid). All experiments in this figure are performed on the CIFAR dataset.

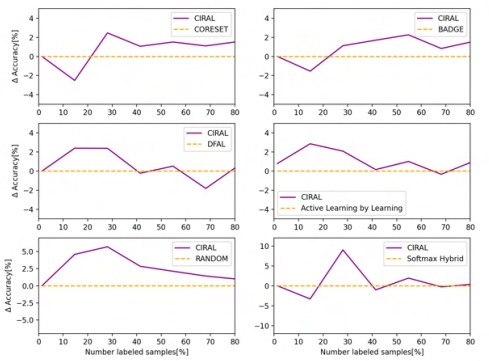
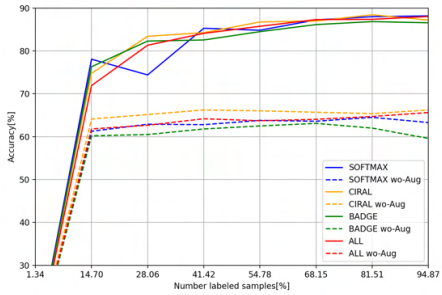
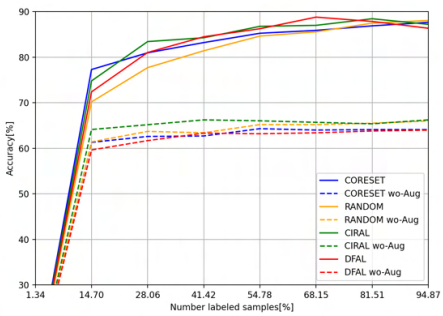


Fig. 5. (LHS) The proposed AL method compared to informative, representative and random methods. (RHS) The proposed AL method compared to other hybrid methods (BADGE, Active Learning by Learning, Softmax Hybrid). All experiments in this figure are performed on the plankton dataset.

method. This result is valid for the other methods as well, suggesting that active learning is effective on the plankton dataset. Furthermore, the results can be studied in more details in figure 5, where the classification accuracy of our method is presented relative to the other methods. In each plot, our method is compared with one other AL method. Similar results can also be found in figure 4, where our method has been applied on the CIFAR dataset. In the latter plot, one can observe that the hybridization benefits from combining informative and representative methods in that it outperforms each of them individually. This performance enhancement compared to the other strategies is a result of incorporating the full feature space while also taking samples close to the inter-class decision boundaries into account. The samples obtained in the latter case help fine-tune the model to gain additional performance. This is particularly evident in figure 5 where one can observe how our proposed method outperforms the coreset representative method when 20% of the samples have been labeled. Moreover, the proposed CIRAL method is also showing promising results compared to the BADGE (Ash et al., 2019), Active Learning by Learning (Hsu and Lin, 2015) and Softmax hybrid (Kaushal et al., 2018) methods. Furthermore, from figure 6 one can observe how the data augmentation module significantly increase the classification accuracy of the active learning methods.

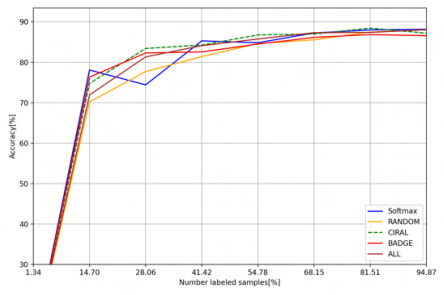


(a) Our method(CIRAL) vs hybrid methods with and without data augmentation on the plankton dataset. Results without augmentation are denoted as 'wo-Aug'

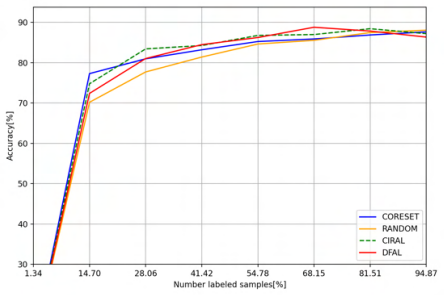


(b) Our method(CIRAL) vs informative/representative with and without data augmentation. Results without augmentation are denoted as 'wo-Aug'

Fig. 6. Comparison of the experimental results with and without data augmentation during training. (a) Performance comparison between our method and other hybrid AL methods with and with out data augmentation. (b) Performance comparison between our hybrid method and informative and representative methods with and without data augmentation.



(a) Our method(CIRAL) vs hybrid methods



(b) Our method(CIRAL) vs informative/representative methods

Fig. 7. Results from the experiment of the proposed method on the plankton dataset. (a) Performance comparison between our method and other hybrid AL methods. (b) Performance comparison between our hybrid method and informative and representative methods.

6. CONCLUSION AND FUTURE WORK

This paper presents a novel framework furthering the field of in-situ underwater planktonic image analysis (Saad et al., 2020, 2021). Manual labeling of planktonic data is time consuming and puts a large burden on the domain experts. The proposed active learning method is able to minimize this effort while still achieving satisfactory classification results. The framework presented in this paper combines metrics for representative and informative sampling and achieve better performance than each of them separately. The method has proven to be efficient on both the benchmark CIFAR dataset and the more complex plankton dataset, suggesting that these metrics should be considered in combination when applying active learning.

The informative metric employed in the proposed framework is dependent on good decision boundaries to get full utilization. Empirical results show that our proposed framework outperforms other state-of-the-art hybrid AL methods. Moreover, the augmentation algorithm which is added as an extension to the originally proposed CIRAL framework Haug et al. (2021), further allowed the model to be more robust on complex data structures that exist in the plankton dataset. An interesting future direction would be to investigate how other representative functions affect the performance of the classifier. In particular, looking at combining Bayesian-based representative metrics with the informative metrics is an interesting direction. Another interesting future direction is to construct, from this novel hybrid AL framework, classifier models that require minimum amount of labeled datasets for training and embedding those created models into AUV platforms for in-situ plankton classification.

ACKNOWLEDGEMENTS

This research is funded by the Research Council of Norway (RCN) IKTPLUSS program (project number 262741) and supported by NTNU AMOS (RCN project number 223254).

REFERENCES

(2015). Plankton imagery data collected from f.g. walton smith in straits of florida from 2014-06-03 to 2014-06-06 and used in the 2015 national data science bowl (nodc accession 0127422). Access: 2020-16-12.

Arthur, D. and Vassilvitskii, S. (2006). k-means++: The advantages of careful seeding. Technical report, Stanford.

Ash, J.T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. (2019). Deep batch active learning by diverse, uncertain gradient lower bounds. *CoRR*, abs/1906.03671. URL http://arxiv.org/abs/1906.03671.

Beluch, W.H., Genewein, T., Nurnberger, A., and Kohler, J.M. (2018). The power of ensembles for active learning in image classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9368–9377. doi:10.1109/CVPR.2018.00976.

Bochinski, E., Bacha, G., Eiselein, V., Walles, T.J., Nejstgaard, J.C., and Sikora, T. (2018). Deep active learning for in situ plankton classification. In *International Conference on Pattern Recognition*, 5–15. Springer.

Ducoffe, M. and Precioso, F. (2018). Adversarial active learning for deep networks: a margin based approach. *CoRR*, abs/1802.09841. URL http://arxiv.org/abs/1802.09841.

Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep bayesian active learning with image data. *CoRR*, abs/1703.02910. URL http://arxiv.org/abs/1703.02910.

Geifman, Y. and El-Yaniv, R. (2017). Deep active learning over the long tail. *CoRR*, abs/1711.00941. URL http://arxiv.org/abs/1711.00941.

Gissin, D. and Shalev-Shwartz, S. (2019). Discriminative active learning. *CoRR*, abs/1907.06347. URL http://arxiv.org/abs/1907.06347.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K.Q. (2017). On calibration of modern neural networks. *CoRR*, abs/1706.04599. URL http://arxiv.org/abs/1706.04599.

Haug, M.L., Saad, A., and Stahl, A. (2021). A combined informative and representative active learning approach for plankton taxa labeling.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385. URL http://arxiv.org/abs/1512.03385.

Hochbaum, D.S. and Shmoys, D.B. (1985). A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2), 180–184.

Hsu, W.N. and Lin, H.T. (2015). Active learning by learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Huang, S., Zhao, J., and Liu, Z. (2018). Cost-effective training of deep cnns with active model adaptation. *CoRR*, abs/1802.05394. URL http://arxiv.org/abs/1802.05394.

Kaushal, V., Sahoo, A., Doctor, K., Uppalapati, N.R., Shetty, S., Singh, P., Iyer, R.K., and Ramakrishnan, G. (2018). Learning from less data: Diversified subset selection and active learning in image classification tasks. *CoRR*, abs/1805.11191. URL http://arxiv.org/abs/1805.11191.

Krishnamurthy, V. (2002). Algorithms for optimal scheduling and management of hidden markov model sensors. *IEEE Transactions on Signal Processing*, 50(6), 1382–1397. doi:10.1109/TSP.2002.1003062.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images.

Lewis, D.D. and Gale, W.A. (1994). A sequential algorithm for training text classifiers. *CoRR*, abs/cmp-lg/9407020. URL http://arxiv.org/abs/cmp-lg/9407020.

Luo, T., Kramer, K., Goldgof, D.B., Hall, L.O., Samson, S., Remsen, A., Hopkins, T., and Cohn, D. (2005). Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6(4).

Mittal, S., Tatarchenko, M., Çiçek, Ö., and Brox, T. (2019). Parting with illusions about deep active learning. *arXiv preprint arXiv:1912.05361*.

Moosavi-Dezfooli, S.M., Fawzi, A., and Frossard, P. (2016). Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2574–2582.

Pastore, V.P., Zimmerman, T.G., Biswas, S.K., and Bianco, S. (2020). Annotation-free learning of plankton for classification and anomaly detection. *Scientific reports*, 10(1), 1–15.

Pinsler, R., Gordon, J., Nalisnick, E., and Hernández-Lobato, J.M. (2021). Bayesian batch active learning as sparse subset approximation.

Ren, P., Xiao, Y., Chang, X., Huang, P.Y., Li, Z., Chen, X., and Wang, X. (2020). A survey of deep active learning.

Saad, A., Bergrum, S., and Stahl, A. (2021). An instance segmentation framework for in-situ plankton taxa assessment. In *Thirteenth International Conference on Machine Vision*, volume 11605, 1160511. International Society for Optics and Photonics.

Saad, A., Stahl, A., Våge, A., Davies, E., Nordam, T., Aberle, N., Ludvigsen, M., Johnsen, G., Sousa, J., and Rajan, K. (2020). Advancing ocean observation with an ai-driven mobile robotic explorer. *Oceanography*, 33(3), 50–59.

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach.

Settles, B. (2009). Active learning literature survey.

Tong, S. and Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov), 45–66.

van der Maaten, L. and Hinton, G. (2008). Viualizing data using t-sne. *Journal of Machine Learning Research*, 9, 2579–2605.

Vodrahalli, K., Li, K., and Malik, J. (2018). Are all training examples created equal? an empirical study. *CoRR*, abs/1811.12569. URL http://arxiv.org/abs/1811.12569.

Wang, K., Zhang, D., Li, Y., Zhang, R., and Lin, L. (2017). Cost-effective active learning for deep image classification. *CoRR*, abs/1701.03551. URL http://arxiv.org/abs/1701.03551.

Yoo, D. and Kweon, I.S. (2019). Learning loss for active learning. *CoRR*, abs/1905.03677. URL http://arxiv.org/abs/1905.03677.

Zhdanov, F. (2019). Diverse mini-batch active learning. *CoRR*, abs/1901.05954. URL http://arxiv.org/abs/1901.05954.

References

[1] 13th ifac conference on control applications in marine systems, robotics, and vehicles. URL https://cams-2021.com.

[2] 13th international conference on digital image processing. URL http://www.icdip.org.

[3] Plankton imagery data collected from f.g. walton smith in straits of florida from 2014-06-03 to 2014-06-06 and used in the 2015 national data science bowl (nodc accession 0127422), 2015. Access: 2020-16-12.

[4] Anaconda software distribution, 2020. URL https://docs.anaconda.com/.

[5] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 2021.

[6] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006.

[7] Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *CoRR*, abs/1906.03671, 2019. URL http://arxiv.org/abs/1906.03671.

[8] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler. The power of ensembles for active learning in image classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9368–9377, 2018. doi: 10.1109/CVPR.2018.00976.

[9] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of*

the IEEE Conference on Computer Vision and Pattern Recognition, pages 9368–9377, 2018.

[10] Erik Bochinski, Ghassen Bacha, Volker Eiselein, Tim JW Walles, Jens C Nejstgaard, and Thomas Sikora. Deep active learning for in situ plankton classification. In *International Conference on Pattern Recognition*, pages 5–15. Springer, 2018.

[11] Antoine Buetti-Dinh, Vanni Galli, Sören Bellenberg, Olga Ilie, Malte Herold, Stephan Christel, Mariia Boretska, Igor V Pivkin, Paul Wilmes, Wolfgang Sand, et al. Deep neural networks outperform human expert's capacity in characterizing bioleaching bacterial biofilm composition. *Biotechnology Reports*, 22:e00321, 2019.

[12] Kashyap Chitta, Jose M. Alvarez, Elmar Haussmann, and Clément Farabet. Less is more: An exploration of data redundancy with active dataset subsampling. *CoRR*, abs/1905.12737, 2019. URL `http://arxiv.org/abs/1905.12737`.

[13] Balázs Csanád Csáji et al. Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, 24(48):7, 2001.

[14] Phil F Culverhouse, Robert Williams, Beatriz Reguera, Vincent Herry, and Sonsoles González-Gil. Do experts make mistakes? a comparison of human and machine indentification of dinoflagellates. *Marine ecology progress series*, 247: 17–25, 2003.

[15] Jialun Dai, Ruchen Wang, Haiyong Zheng, Guangrong Ji, and Xiaoyan Qiao. Zooplanktonet: Deep convolutional network for zooplankton classification. In *OCEANS 2016 - Shanghai*, pages 1–6, 2016. doi: 10.1109/OCEANSAP.2016.7485680.

[16] Melanie Ducoffe and Frédéric Precioso. Adversarial active learning for deep networks: a margin based approach. *CoRR*, abs/1802.09841, 2018. URL `http://arxiv.org/abs/1802.09841`.

[17] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[18] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. *CoRR*, abs/1703.02910, 2017. URL `http://arxiv.org/abs/1703.02910`.

[19] Utkarsh Gaur, Matthew Kourakis, Erin Newman-Smith, William Smith, and BS Manjunath. Membrane segmentation via active learning with deep networks.

In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1943–1947. IEEE, 2016.

[20] Yonatan Geifman and Ran El-Yaniv. Deep active learning over the long tail. *arXiv preprint arXiv:1711.00941*, 2017.

[21] Daniel Gissin and Shai Shalev-Shwartz. Discriminative active learning. *CoRR*, abs/1907.06347, 2019. URL `http://arxiv.org/abs/1907.06347`.

[22] Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E. Turner. Meta-learning probabilistic inference for prediction, 2019.

[23] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. *CoRR*, abs/1706.04599, 2017. URL `http://arxiv.org/abs/1706.04599`.

[24] Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In *NIPS*, pages 593–600. Citeseer, 2007.

[25] Martin Lund Haug. Applying active-learning techniques in machine learning to minimize labeling effort, 2020.

[26] Martin Lund Haug, Aya Saad, and Annette Stahl. A combined informative and representative active learning approach for plankton taxa labeling. 2021.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[28] Dorit S Hochbaum and David B Shmoys. A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2):180–184, 1985.

[29] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[30] Wei-Ning Hsu and Hsuan-Tien Lin. Active learning by learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[31] Shell Xu Hu, Pablo G. Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil D. Lawrence, and Andreas Damianou. Empirical bayes transductive meta-learning with synthetic gradients, 2020.

[32] Sheng-Jun Huang, Jia-Wei Zhao, and Zhao-Yang Liu. Cost-effective training of deep cnns with active model adaptation. *CoRR*, abs/1802.05394, 2018. URL http://arxiv.org/abs/1802.05394.

[33] Vishal Kaushal, Anurag Sahoo, Khoshrav Doctor, Narasimha Raju Uppalapati, Suyash Shetty, Pankaj Singh, Rishabh K. Iyer, and Ganesh Ramakrishnan. Learning from less data: Diversified subset selection and active learning in image classification tasks. *CoRR*, abs/1805.11191, 2018. URL http://arxiv.org/abs/1805.11191.

[34] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *CoRR*, abs/1906.08158, 2019. URL http://arxiv.org/abs/1906.08158.

[35] V. Krishnamurthy. Algorithms for optimal scheduling and management of hidden markov model sensors. *IEEE Transactions on Signal Processing*, 50(6):1382–1397, 2002. doi: 10.1109/TSP.2002.1003062.

[36] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[37] Hae Beom Lee, Hayeon Lee, Donghyun Na, Saehoon Kim, Minseop Park, Eunho Yang, and Sung Ju Hwang. Learning to balance: Bayesian meta-learning for imbalanced and out-of-distribution tasks, 2020.

[38] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. *CoRR*, abs/cmp-lg/9407020, 1994. URL http://arxiv.org/abs/cmp-lg/9407020.

[39] Alessandra Lumini, Loris Nanni, and Gianluca Maguolo. Deep learning for plankton and coral classification. *Applied Computing and Informatics*, 2020.

[40] Tong Luo, Kurt Kramer, Dmitry B Goldgof, Lawrence O Hall, Scott Samson, Andrew Remsen, Thomas Hopkins, and David Cohn. Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6(4), 2005.

[41] Sudhanshu Mittal, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox. Parting with illusions about deep active learning. *arXiv preprint arXiv:1912.05361*, 2019.

[42] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[43] Prateek Munjal, Nasir Hayat, Munawar Hayat, Jamshid Sourati, and Shadab Khan. Towards robust and reproducible active learning using neural networks. *ArXiv, abs/2002.09564*, 2020.

[44] Hariank Muthakana. *Uncertainty and diversity in deep active image classification.* PhD thesis, Carnegie Mellon University Pittsburgh, PA, 2019.

[45] Vito P Pastore, Thomas G Zimmerman, Sujoy K Biswas, and Simone Bianco. Annotation-free learning of plankton for classification and anomaly detection. *Scientific reports*, 10(1):1–15, 2020.

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python, 2011.

[48] Robert Pinsler, Jonathan Gordon, Eric Nalisnick, and José Miguel Hernández-Lobato. Bayesian batch active learning as sparse subset approximation, 2021.

[49] Remus Pop and Patric Fulop. Deep ensemble bayesian active learning : Addressing the mode collapse issue in monte carlo dropout via ensembles. *CoRR*, abs/1811.03897, 2018. URL `http://arxiv.org/abs/1811.03897`.

[50] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. A survey of deep active learning, 2020.

[51] Simon-Martin Schröder, Rainer Kiko, and Reinhard Koch. Morphocluster: Efficient annotation of plankton images by clustering. *Sensors*, 20(11):3060, 2020.

[52] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach, 2018.

[53] Burr Settles. Active learning literature survey. 2009.

[54] Aditya Siddhant and Zachary C Lipton. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. *arXiv preprint arXiv:1808.05697*, 2018.

[55] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. *arXiv preprint arXiv:1904.00370*, 2019.

[56] Shiliang Sun, Honglei Shi, and Yuanbin Wu. A survey of multi-source domain adaptation. *Information Fusion*, 24:84–92, 2015.

[57] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov): 45–66, 2001.

[58] Laurens van der Maaten and Geoffrey Hinton. Viualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 11 2008.

[59] Joannes Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.

[60] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[61] Kailas Vodrahalli, Ke Li, and Jitendra Malik. Are all training examples created equal? an empirical study. *CoRR*, abs/1811.12569, 2018. URL `http://arxiv.org/abs/1811.12569`.

[62] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *CoRR*, abs/1701.03551, 2017. URL `http://arxiv.org/abs/1701.03551`.

[63] Xing Wu, Cheng Chen, Mingyu Zhong, Jianjia Wang, and Jun Shi. Covid-al: The diagnosis of covid-19 with deep active learning. *Medical Image Analysis*, 68: 101913, 2021.

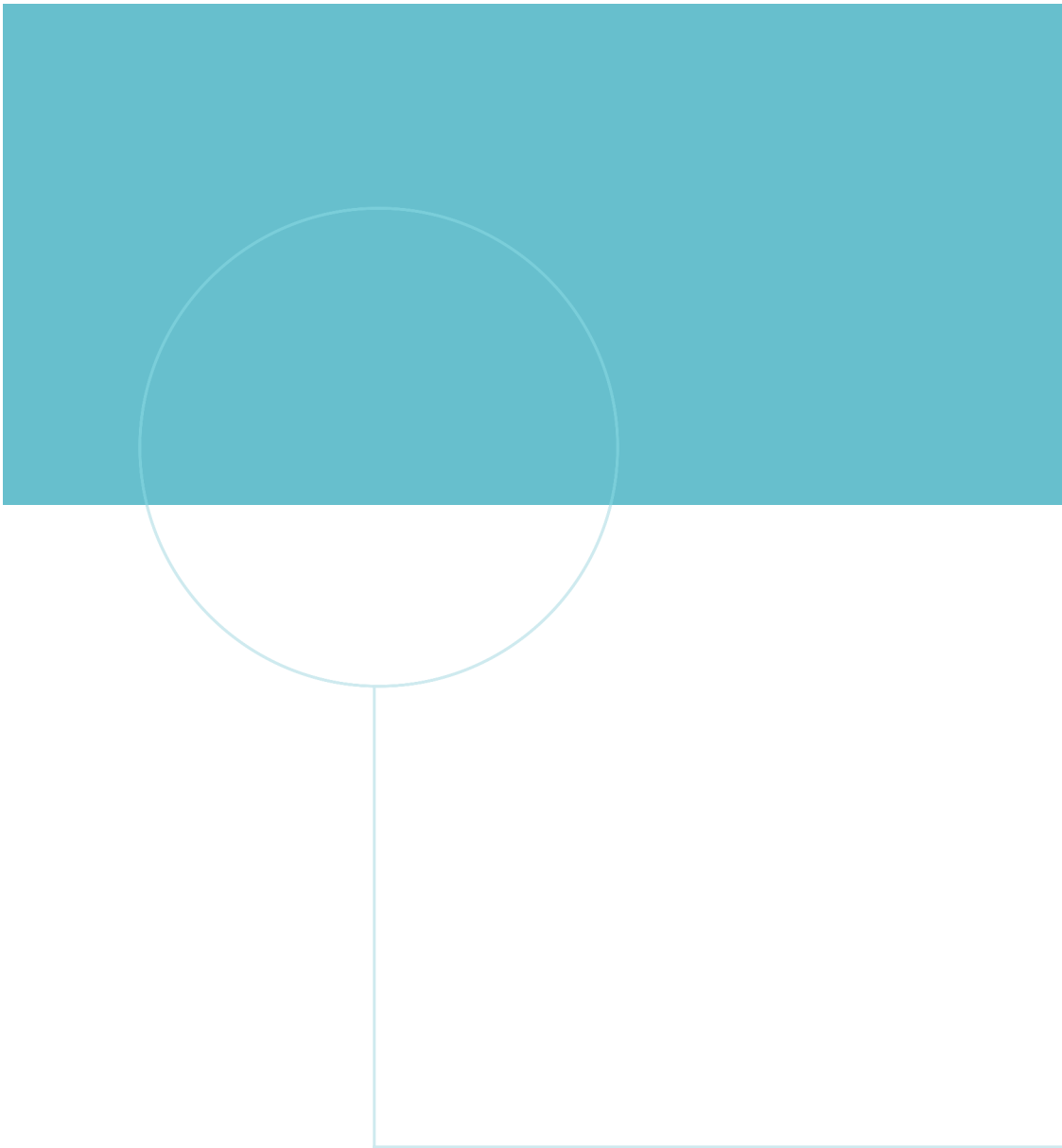
[64] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 399–407. Springer, 2017.

[65] Donggeun Yoo and In So Kweon. Learning loss for active learning. *CoRR*, abs/1905.03677, 2019. URL `http://arxiv.org/abs/1905.03677`.

[66] Feng Zhao, Feng Lin, and Hock Soon Seah. Binary sipper plankton image classification using random subspace. *Neurocomputing*, 73(10-12):1853–1860, 2010.

[67] Fedor Zhdanov. Diverse mini-batch active learning. *CoRR*, abs/1901.05954, 2019. URL `http://arxiv.org/abs/1901.05954`.

[68] Haiyong Zheng, Ruchen Wang, Zhibin Yu, Nan Wang, Zhaorui Gu, and Bing Zheng. Automatic plankton image classification combining multiple view features via multiple kernel learning. *BMC bioinformatics*, 18(16):1–18, 2017.



NTNU

Norwegian University of
Science and Technology