

Mikko Impiö

**SEMI-SUPERVISED LEARNING IN HABITAT
CLASSIFICATION FROM REMOTELY-SENSED
IMAGERY**

Master of Science Thesis
Faculty of Information Technology and Communication Sciences
Examiners: Dr. Jenni Raitoharju and Dr. Laura Uusitalo
April 2022

ABSTRACT

Mikko Impiö: Semi-supervised learning in habitat classification from remotely-sensed imagery
Master of Science Thesis
Tampere University
Master's Programme in Computing Sciences
April 2022

Remote sensing helps monitor and evaluate the state of ecosystems, covering also wilderness areas that can be hard to access for field observations. Wilderness areas, such as the ones in northern Lapland, are home to endangered species and habitat types. Automatic detection and classification of habitats is a difficult task, as target class distributions are long-tailed, fine-grained, and have semantic properties that can be difficult to distinguish even for humans and especially from limited remotely sensed imagery. Training data for building models is often sparse, point-like, and limited to areas accessible by foot. This thesis presents methods for habitat classification from limited data using supervised, unsupervised, and semi-supervised methods. The presented approaches take advantage of the large amounts of unannotated and weakly annotated source data that is available. Convolutional neural networks and random forests are compared and an ensemble model combining both approaches is shown to increase classification performance. Convolutional neural networks are also used to produce fully unsupervised segmentation maps. The classification and segmentation maps are produced for the entire northern Lapland area.

Keywords: semi-supervised learning, deep learning, land cover classification, remote sensing, clustering

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

TIIVISTELMÄ

Mikko Impiö: Puoliohjattu koneoppiminen kaukokartoitusaineistoista tehtävässä luontotyyppien tulkinnassa
Diplomityö
Tampereen yliopisto
Tietotekniikan DI-ohjelma
Huhtikuu 2022

Ekosysteemien tilaa voidaan seurata ja arvioida kaukokartoituksen avulla kenttähavaintoja tehokkaammin, etenkin laajoilla ja vaikeasti saavutettavilla erämaa-alueilla. Esimerkiksi Pohjois-Lapissa elää uhanalaisia lajeja ja luontotyyppejä, joiden suojelu on tärkeää. Luontotyyppien automaattinen tulkinta on hankala tehtävä, sillä luokkia on yleensä paljon, ne ovat hyvin samantlaisia keskenään ja niiden erottaminen toisistaan voi olla vaikeaa jopa asiantuntijalle maastossa. Kaukokartoitusaineistojen, eli satelliiteista ja ilmasta tehtyjen havaintojen avulla, luokittelu muuttuu vielä hankalammaksi. Luokittelumallien muodostamiseen kerättävä kenttäaineisto on yleensä pistemäistä ja rajoittuu yleensä ihmisen saavutettavissa oleviin alueisiin. Tässä opinnäytetyössä tutkitaan koneoppimismenetelmiä kaukokartoitusaineistosta tehtävään luontotyyppien luokitteluun, hyödyntäen valvottuja, valvomattomia ja puolivalvottuja koneoppimismenetelmiä. Työssä hyödynnetään saatavilla olevia suuria määriä annotoitua ja rajoitetusti annotoitua kaukokartoitusmateriaalia. Työssä verrataan etenkin konvolutiivisia neuroverkkoja (convolutional neural network, CNN) ja satunnaismetsiä (random forests), sekä esitellään molempia lähestymistapoja yhdistävä uusi menetelmä, jonka näytetään myös tuottavan aiempaa luotettavampia luokittelutuloksia. Konvolutiivisia neuroverkkoja käytetään myös täysin ohjaamattoman segmentointikartan tuottamiseen. Työssä tuotetaan luokitus- ja segmentointikartat koko Pohjois-Lapin alueelle, joka on uutta tämänkaltaisella aineistolla ja mallilla.

Avainsanat: puoliohjattu koneoppiminen, syväoppiminen, maanpeitetulkinta, kaukokartoitus, klusterointi

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

PREFACE

Tämä diplomityö on tehty Suomen ympäristökeskus SYKelle osana Metsähallituksen koordinoimaa Ylä-Lapin kaukokartoitushanketta, jonka tavoite on päivittää tietoa Ylä-Lapin biotoopeista hyödyntäen satelliitti- ja laserkeilausaineistoja. Valtava määrä luontotyyppejä on katoamisuhan alla, ja toivon että tämä työ on pieni pisara siinä tutkimusten virrassa, josta on apua tämän kehityskulun kääntämisessä.

Haluan kiittää etenkin työn ohjaajaa, Jenni Raitoharjua, jonka asiantuntemuksen ansiosta olen oppinut uutta ja kehittynyt tutkijana projektin aikana. Kiitän myös projektin toista ohjaajaa Saku Anttilaa, joka antoi tämän mahdollisuuden astua kaukokartoituksen kiehtovaan maailmaan.

Kiitän avusta Pekka Härmää ja muita kaukokartoitusprojektin asiantuntijoita. Työn tekemisessä auttoivat erinomaiset lähtödatat, joiden tuottamiseen en olisi itse kyennyt. Kiitokset kuuluvat myös Kristian Meissnerille sekä SYKEN kollegoilleni Uudistuvan ympäristötiedon strategisessa ohjelmassa. Ammatilliseen kehitykseeni ovat vaikuttaneet suunnattomasti myös kollegat edellisissä työpaikoissani. Kiitokset siis myös heille.

Tämän työn myötä päättyvät myös opintoni Tampereen (teknillisessä) yliopistossa. Opiskeluaikani ovat olleet elämäni parasta aikaa ja olen siitä kiitollinen ystävilleeni, joiden kanssa olen saanut jakaa nämä vuodet. Kiitokset niin sanotulle BFI:lle unohtumattomista seikkailuista ja vertaistuesta. Sähkökillan opintonurkkauksessa vietetyt tunnit lasketaan luultavasti tuhansissa. Niiden muistot ovat kullanneet ystävät killassa, jonka toiminnassa olen saanut tiiviisti olla mukana.

Lopuksi haluan kiittää vielä perhettäni, ja rakasta puolisoani Nooraa: viisainta, hauskinta ja lämminsydämisintä ihmistä jonka tunnen. Ilman sinua tämä työ ei olisi valmistunut koskaan.

Helsingissä, 27th April 2022

Mikko Impiö

CONTENTS

1.	Introduction	1
2.	Background	6
2.1	Supervised learning	6
2.1.1	Semantics	7
2.1.2	Random forests	8
2.1.3	Convolutional neural networks	9
2.2	Information theory	10
2.3	Unsupervised and semi-supervised learning	12
2.3.1	Invariant information clustering	13
2.3.2	Noisy student training	14
2.4	Remote sensing	15
3.	Prior work	18
4.	Proposed method	22
5.	Data	26
5.1	Annotations	26
5.2	Remote sensing data	29
6.	Experimental setup	31
6.1	Data	31
6.2	Machine learning setup	32
6.3	Evaluation metrics	33
7.	Results	35
7.1	Classification	35
7.1.1	Sensitivity studies	49
7.1.2	Classification maps	54
7.2	Unsupervised segmentation	65
8.	Conclusion	69
	References	71
	Appendix A: Source data	78
	Appendix B: Full result tables	83
	Appendix C: Additional figures	93

LIST OF SYMBOLS AND ABBREVIATIONS

CNN	Convolutional neural network
GCS	General Classification System for Finland's biotopes
IIC	Invariant information clustering
LiDAR	Light detection and ranging
MLP	Multilayer perceptron
NDVI	Normalized difference vegetation index
NIR	Near infrared
OBIA	Object-based image analysis
RGB	Red, green and blue color channels
ROC	Receiver operating characteristics
TTA	Test-time augmentation
VRE	Vegetation red edge

1. INTRODUCTION

Observations of our natural environment show that global biodiversity is in decline. Short-term extinction rates are among the highest in our planets history, with global biodiversity loss projections ranging from 10%-75% for the next century [1, 2, 3]. There is evidence that most of the loss of species is due to human action [2, 4, 5], making conservation of the environment a top priority for the humankind.

The biodiversity crisis we are facing is not only visible in species loss, but also in biotope and habitat loss. Habitats, referring to local communities of species and the environment they inhabit, are also being destroyed and degraded due to human actions and climate change [4, 6]. In addition to their intrinsic value, these complex communities of species provide important *ecological services*: they produce oxygen and food, capture carbon dioxide, filter water, and retain it locally, for example. Other species, including humans, benefit from these services, entangling all species into a complex web of interdependencies where the loss of diversity can have chaotic effects on the whole system. [3, 5, 7]

Habitat monitoring is important to better understand the impacts and causes of biodiversity loss. The main methods for habitat monitoring are on-site field surveys and remote sensing [8]. Field surveys provide high quality local information, but they are expensive and time-consuming, especially in remote areas [9]. Remote sensing can collect a lot of data over a large area at low cost, but the nature of the data collected is very different from the field survey data. Some information that is only available in-situ is always lost. A study by Rhodes et al. [10] compared remote sensed data to field surveys in classification for bird abundances and habitats, measuring a relative explanatory power of 73% for remote sensed data compared to field measurements. Due to these different strengths, field surveys and remote sensing are complementary to each other, where field surveys are commonly used to produce ground truth data for large-scale modeling done from remote sensed data [11, 12, 13, 14, 15].

Vast amounts of remote sensing imagery data are available, thanks to Earth observation satellites, such as the Sentinel and Landsat series. These satellites produce daily observations of the Earth using different instruments, such as multispectral cameras and synthetic aperture radars (SAR) [16]. These instruments produce large image rasters, often

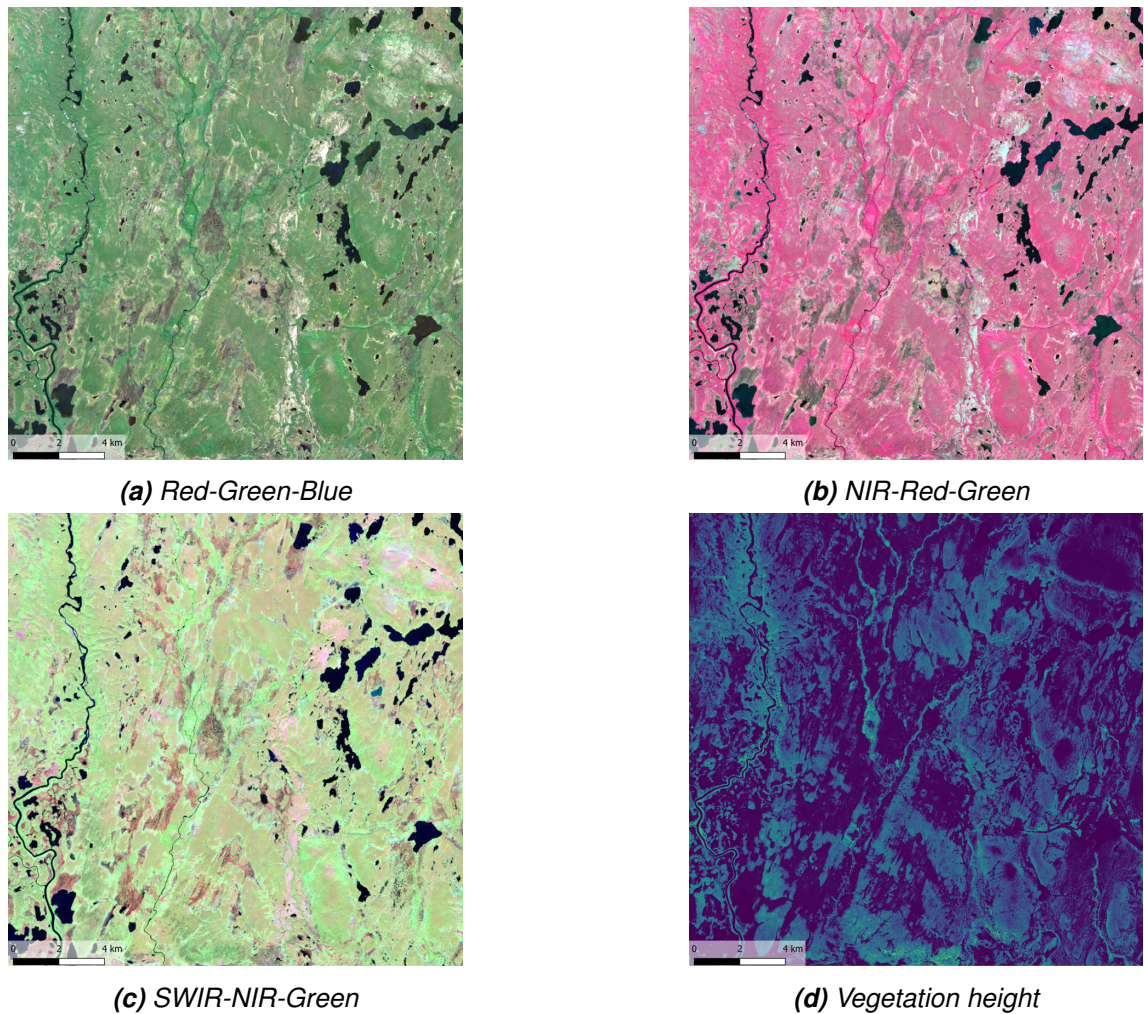


Figure 1.1. False color rasters (a-c) from the multispectral camera aboard the Sentinel-2 satellite and vegetation height (d) calculated from a laser scanning flight over the same area. The false-color images are generated by selecting specific wavelengths from a multispectral camera for the different channels of the RGB image. The subfigure captions specify which spectral bands were used for the RGB channels.

with a spatial resolution of over 10m, meaning that each pixel corresponds to a $10\text{m} \times 10\text{m}$ area in nature. Aerial missions can provide additional higher resolution data, for example laser scanned point clouds that can be processed into height maps and canopy cover rasters. Examples of multispectral and laser scanned data rasters can be seen in Figure 1.1.

Fell habitats endemic to northern Europe have a high priority in conservation interest. Climate change has the greatest effect near the poles, and the changes caused by the current mean temperature rise can already be seen in Lapland's ecosystems [17]. Mires with permafrost are melting and southern species are pushing northwards. A large amount of fell habitats are already endangered and on the Red List of habitats [17], making monitoring their changes urgent and important. The study area of this thesis is delimited to the most northern Lapland, as shown in Figure 1.2.

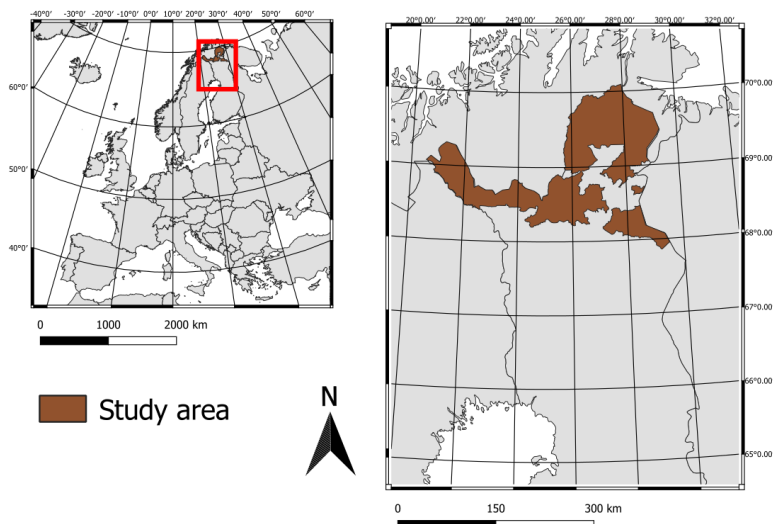


Figure 1.2. The study area in northern Lapland.

This thesis uses remote sensed imagery data and habitat information collected on field to automatically classify land cover to discrete categories. These categories are different biotopes or habitat types, describing the assemblage of vegetation and species in the area. The main focus is on different machine learning methods that can be used to produce a model $f : \mathcal{X} \rightarrow \mathcal{Y}$ that maps input \mathcal{X} to a target class \mathcal{Y} . In this case, the input is remote sensing imagery and the target class is a biotope or habitat class. These models are usually trained using only pairs of training data $(x, y) \in \mathcal{X} \times \mathcal{Y}$, without prior assumptions of the system, contrary to Bayesian approaches.

Most of the research done in this thesis concerns methods applied to *neural networks*, which are usually characterized as over-parameterized non-linear models that are trained by iteratively optimizing a loss-function in the parameter space. Deep learning methods usually need a large amount of data to train models. In the problem of using remote sensed data for habitat classification, an interesting disparity arises where the amount of input data is abundant, but target ground truth annotations are scarce and difficult to collect. In addition, vast amounts of out-of-domain annotations are available, for example for land-cover classification of different taxonomies. Transfer learning, unsupervised learning, and semi-supervised learning can utilize these larger datasets for training the models.

Semi-supervised methods combine the target of approximating a function to categorize input data and learning additional information from input data without a target annotation. In this thesis, recent research on semi- and unsupervised learning is applied on the habitat classification problem, with the main approaches being *noisy student training* proposed by Xie et al. [18], and *invariant information clustering* (IIC) proposed by Ji et al. [19]. In contrast to supervised learning models, where the target set is known,

the goal in unsupervised learning is to find a mapping from input space to N different output classes, or *clusters*. It is often desired to produce clusters that have large differences between clusters, but small differences among the samples inside a cluster. Semi-supervised methods attempt to produce a similar mapping as supervised methods, but with methods that make use of data that is unlabeled, i.e samples from the input set \mathcal{X} without the pair from \mathcal{Y} .

The semi- and unsupervised methods are compared to more common transfer learning approaches, where a neural network is first trained with annotated data not necessarily in the domain of the final classification. The pre-trained model learns general representation mappings, which can be used when training a final classification model in the target labeling domain. The vast availability of out-of-domain annotations from land cover datasets makes transfer-learning an ideal approach to the problem of fine-grained habitat classification.

When habitats are classified in the field, the classification usually applies to a small area around a sampling point. Remote sensing imagery has a low spatial resolution, making field observations essentially point annotations for single pixels in the raster. Land cover classification methods are often split to two main approaches: object-based and pixel-based classification [20, 21]. Pixel-based classification classifies each pixel separately, while object-based image analysis (commonly referred to as OBIA) categorizes a larger area spanning over several pixels. Due to the low spatial resolution of remote sensed imagery, pixel-based methods are common [11, 12, 14, 22], as it is reasonable to consider each pixel as a separate object to be classified. Object-based approach considers a larger area around the pixel, with common approaches being using unsupervised segmentation to find homogeneous areas and classifying them based on their contents [21, 23], or classifying areas on a coarse level, separating the "objects", and then classifying them to a finer taxonomy [24, 25]. Modern approaches with convolutional neural networks (CNNs) performing semantic segmentation on remote sensed imagery [13, 15, 26, 27] can be considered object-based, since the methods use the local information around each pixel that is to be semantically classified to a specific class.

This thesis proposes a method combining the two approaches to one by ensembling two separate models to classify each pixel. Each pixel is classified using both a random forest (RF) model and a CNN, the former using only the pixel values for classification while the latter also uses the surrounding area to support classification. It is shown that this ensemble approach performs better than the classifiers on their own. This thesis also proposes using a special random cropping augmentation for a problem like this, where only the center pixel of an image is classified. Figure 1.3 summarises the different approaches to classification and segmentation and how the proposed method compares to them.





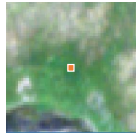
Method	Source data	Ground truth label
Image classification	 N -channel image	"River" Image-level label
Image segmentation	 N -channel image	 Segmentation map
Single-pixel classification	 N -dimensional vector representing channel values	"Birch forest" Pixel-level label
Proposed method	 N -channel image and N -dimensional vector representing center pixel channel values	"Birch forest" Pixel-level label

Figure 1.3. The proposed approach of combining pixel-based classification with image-level classification

The main contributions of this thesis are:

- Fine-grained classification of fell habitat types from remote-sensed imagery for two different class taxonomies
- Presenting a classification approach using an ensemble of CNNs and random forests, combining pixel-based and object-based classification approaches
- Using sparse pixel annotations with a large unannotated dataset for semi-supervised learning
- Proposing a crop augmentation method for surrounding-area-aware center pixel classification
- Applying fully unsupervised segmentation using CNNs for land cover segmentation

This thesis is divided to eight chapters. This introduction chapter introduces the problem and outlines the main contributions of this thesis. Necessary background for supervised learning, algorithms, training approaches, and remote sensing is discussed in Chapter 2. Chapter 3 ties this thesis to the prior research on land cover classification, habitat mapping, and machine learning. The proposed approaches are discussed in depth in Chapter 4. Chapters 5 and 6 describe the data sources used in the study and the technical details of the experiments. Chapter 7 shows and analyzes the results. The results are divided to cross-validated classification results, sensitivity studies, and qualitative results illustrating the final classification maps produced. Finally, 8 concludes the thesis. The appendix contains additional figures and references that are omitted from the main text for simplicity and legibility.

2. BACKGROUND

This chapter discusses the necessary background for this thesis. The proposed method focuses on machine learning methods, thus supervised learning and different training approaches and algorithms are discussed the most. Supervised learning in general is discussed along with the semantic problems that arise from classifying complex objects to discrete classes. Convolutional neural networks and random forests are also introduced.

The unsupervised and semi-supervised learning approaches are based on optimization approaches that rely on different loss functions. These loss functions, categorical cross-entropy and mutual information are introduced, and the unsupervised and semi-supervised approaches used in this thesis are discussed. Finally, background on the remote sensing data used in this work is discussed.

2.1 Supervised learning

Supervised learning methods attempt to learn a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ from an input set \mathcal{X} to a target set \mathcal{Y} using input-output pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$ as *training data*. In theory, the input and target pairs can be any objects that can be combined so that each element in the input set corresponds to a single element in the target set. In the case of a discrete target set, the problem is called *classification* and with a continuous target it is called *regression*. [28]

For example, the input set could be the set of all 64×64 grayscale images with values in the range $[0, 255]$, and the target set could be $\mathcal{Y} = \{\text{"Forest"}, \text{"Not forest"}\}$. The resulting function $f(x)$ would be a *classifier* that classifies images to two categories based on the images' semantic content. With a *binary classification problem* like this, the function output could be 1 in the case of a forest image and 0 in the case of a non-forest image. Now, if a grayscale image is denoted in vector form as \mathbf{x} the function would be

$$f(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \text{ semantically contains a "forest"}. \\ 0, & \text{otherwise.} \end{cases} \quad (2.1)$$

In this thesis, the target set \mathcal{Y} with discrete and categorical values will be called a *taxonomy*. This differs slightly from the traditional use of the word since a taxonomy usually

assumes that the set is hierarchical. The set {"Forest", "Not forest"} is not hierarchical, but most other classification taxonomies presented in this thesis will be. The cardinality of \mathcal{Y} , or the number of classes, will be marked as C . In the above example, $C = 2$.

The number of *features* in the input vector \mathbf{x} is the number of dimensions D in the input space $\mathcal{X} \subseteq \mathbb{R}^D$. Each pixel of a 64×64 grayscale image is a separate feature. In this case, when the grayscale values are discrete, the input space $\mathcal{X} \subseteq \{0, 1, \dots, 255\}^{64 \times 64}$ is a non-infinite but extremely large space of all possible grayscale images of this size and bit depth. A high-dimensional input space is difficult to map to a smaller target space. Methods like convolutional neural networks (CNNs), that reduce the dimensionality in different stages by convolving local features and pooling them together, are extremely efficient in mapping high-dimensional data structured in a lattice, such as an image. On the opposite side of input space dimensions is single pixel classification. If each pixel is classified separately, the number of input features is $D = 1$, producing a function from 255 possible pixel values to C class values. If the image has multiple channels, the input space dimension D increases. This thesis approaches both types of input spaces, full image classification by classifying an entire view to a single class with CNNs, and single pixel classification by handling each pixel separately with random forests.

2.1.1 Semantics

The forest taxonomy specified above can be seen as collectively exhaustive, since it contains all possible semantic contents for a grayscale image. If we ignore the possible vagueness of the concept "forest", every image $\mathbf{x} \in \mathcal{X}$ represents either a "forest" or "not forest". In practice, classification taxonomies are rarely collectively exhaustive. It is possible to build a reliable classifier with a target set taxonomy of {"Forest", "Lake"} that will work whenever the image \mathbf{x} represents either a forest or a lake, but will output "Forest" or "Lake" also when the semantic meaning of the image is something completely different. For this reason, classifiers dealing with semantic categories are rarely mathematically perfect functions, mapping each element from the input set into an output set, at least without explicit boundaries on the input set.

It is easy to see that setting a threshold between the elements in the set {"Forest", "Not forest"} resembles the "sorites paradox" [29], in the form of the question: "How many trees does make a forest?". Where is the line where a bunch of trees turns into a forest? This vagueness problem has been studied in philosophy [30] and is often ignored in machine learning and classification problems, although it is present almost always when classes do not have explicitly defined boundaries. As machine learning models attempt to learn these boundaries from limited data, the resulting classification boundary can be interpreted as incorrect by some observers, even though it would reflect the best possible boundary that can be learned.

The problem of semantics and vagueness is well present in habitat classification. Ecosystems and habitats are complex systems composed of different species and gradually change from one to another. Attempts to produce categorical differences between habitats is done mainly from human perspective and has large cultural differences. For example, the general classification system for Finland's biotopes used in this thesis contains categorizations for wetlands that are mostly used only in Finland. These wetlands can change classes depending on the amount of water they contain, making the classification between classes vague for even experts.

2.1.2 Random forests

Decision tree -based methods are commonly used in supervised statistical learning. These methods recursively partition the feature space into binary regions, resulting in a rectangular partition with decision boundaries perpendicular to the basis vectors of the feature space. Traditional linear models often fail in scenarios where the relationship between inputs and targets are non-linear, and decision trees can model these complex nonlinear relationships in the data. [31]

With an input set $\mathcal{X} \subseteq \mathbb{R}^D$ and a target set \mathcal{Y} , the feature space is split recursively using a binary threshold on one of the D features, where the split produces a model with the best explanatory response on the target \mathcal{Y} . The resulting two splits of the feature space are then split recursively until the partition of the feature space is able to predict the target response from the training set. [31]

A decision tree is grown by recursively choosing the best feature and a value for that feature to produce splits of the feature space into M regions R_1, R_2, \dots, R_M . For example, at first the training dataset of N samples of input-target pairs $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ is used to produce two regions, R_1 and R_2 . For each region value $m = 1, 2$ and class c , a proportion of class observations p_{mc} in the split is calculated as

$$p_{mc} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = c), \quad (2.2)$$

where the summation is over samples assigned to split R_m , N_m is the number of such samples, and $I(y_i = c)$ gets a value of 1 if the target value of x_i is a certain class. All observations are classified into the majority class in the split, $\operatorname{argmax}_c p_{mc}$. Each region then gets a measure of *node impurity* Q_m , which is commonly calculated using entropy as

$$Q_m = - \sum_{c=1}^C p_{mc} \log(p_{mc}) \quad (2.3)$$

or the Gini index as

$$Q_m = \sum_{c=1}^C p_{mc}(1 - p_{mc}), \quad (2.4)$$

corresponding to the variance across classes. The feature and the splitting point is chosen so that the node impurity in both splits is minimized. [31] For each split, a subsplit can be calculated, the node impurity is minimized across all new splits. To prevent overfitting, a pruning approach is used. First, a full tree that explains the full dataset without error is built. The most recent splits where increase in error is the smallest are removed. [28]

A decision tree produces an easily interpretable model and is fast to train. However, a single tree is very unstable and slight changes in the input data can change large areas of the resulting model. A solution for this problem is "bagging", where several decision trees are trained with subsets of the full dataset. During inference, a majority vote of the classification is performed on this ensemble of decision trees. Training decision trees on the same dataset can produce highly correlated models, reducing the effect of bagging. Therefore, random forest attempts to decorrelate the decision trees by choosing also subsets of features that are used for training. The resulting models are fairly robust and produce good results also on small and imbalanced datasets. [28]

2.1.3 Convolutional neural networks

Convolutional neural networks (CNNs) are a family of neural networks that are based on using the convolution operation. Convolution is a very common operation in signal processing and has many applications, such as different filtering operations. If we have a signal $x(t)$, for every t we can calculate the convolution output $s(t)$ using a *kernel* w :

$$s(t) = \int x(a)w(t - a)da, \quad (2.5)$$

making the convolution in essence a weighted sum of all input values for each point in t . [32]

For a two-dimensional discrete image, the kernel function w is often finite with a height and width of M and N . Convolution for discrete and multi-dimensional functions, such as an image I , is essentially a sliding window that calculates the weighted sum of the input pixel $I(i, j)$ and its surroundings:

$$S(i, j) = \sum_m^M \sum_n^N I(i - m, j - n)w(m, n), \quad (2.6)$$

where $I(i, j)$ is the pixel value in position (i, j) and $w(m, n)$ is the kernel weight [32]. Often the ranges of m and n are symmetrically around zero, e.g. -2,-1,0,1,2, centering

the kernel window on the image pixel. In practice, convolutions use other tricks such as kernel flipping, border handling and other computational operations for optimizing the calculations [32].

CNNs work well for images because using the convolutional kernel approach leads to *parameter sharing* and *representation equivariance*. Parameter sharing, or using the same kernel functions for all image pixels, means that much less parameters are needed to learn. The learned kernels activate on similar features everywhere in the image. Together with *pooling*, decimating the input based on choosing for example the maximum value in a neighborhood, the outputs of CNNs are invariant to location changes of features in images. A image of a cat will produce the same output regardless of its position in the image, resulting in representation equivariance.

A common CNN architecture is the ResNet [33] family. In addition to convolutional and pooling layers, the ResNet networks apply *batch normalization* layers and connection skipping. This allows reducing internal covariance shift and problems with vanishing gradients during optimization [33]. CNNs can have a varying amount of layers, for example, ResNet18 contains 18 layers and ResNet50 50 layers. The useful amount of layers often correlates with the amount of training data available, but is often chosen with empirical tests. Smaller networks have less parameters, making them faster to train and perform classifications. Like other neural networks, CNNs have nonlinear functions between layers and are usually optimized using gradient-based optimization methods.

The output of the convolutional and pooling layers is usually a single-dimensional vector. Commonly this vector is referred to as the *representation vector*, as it represents the visual content of an image. The final layers of a CNN are usually *fully connected layers*, or a *multilayer perceptron* (MLP), that produce a nonlinear function of this representation vector into a desired output format. This could be for example a class distribution vector or a single value. Sometimes it is useful to conceptually separate the convolutional layers and the MLP into a *CNN backbone* that maps images to a representation vector, and a *classification head* that does the final output mapping.

2.2 Information theory

Most machine learning methods depend on concepts formalized in information theory. CNNs are usually trained by optimizing a loss function, that often is *categorical cross-entropy* or a derivation of it. The methods used in this thesis uses cross-entropy and well as *mutual information* as loss functions. The necessary background for these concepts is explained in this section.

If we have a discrete random variable X with a probability distribution $p = P(X)$, the *entropy* of X is

$$H(p) = - \sum_i p_i \log(p_i) \quad (2.7)$$

where p_i is the probability of outcome x_i [28]. The unit of base-2 entropy is *bits* and is often referred to as Shannon entropy, and is the average amount of information needed in order to encode a message consisting of the outcomes of the random variable X [34]. Entropy can be seen as the amount of "disorder" in a distribution, as a uniform distribution maximizes the entropy function.

The entropy of a message, for example an 8-bit, $N \times M$ grayscale image, can be found by calculating a probability distribution across the possible values each message "letter" or a pixel can have. The possible pixel values are a set $\Omega = \{0, 1, \dots, 255\}$. Each pixel value is sampled from this set and placed to the grid to form an image. The probability of a pixel getting a value p_i is defined by the distribution p . In what we consider natural images, some grayscale values are more probable than others, as images contain for example uniformly colored surfaces and high contrast edges. Thus the distribution p for a natural images is usually not uniform and has a lower entropy than what a image sampled from a uniform distribution would have. It can be said that the pixel value distribution that characterizes an image *contains information*, if the entropy is lower than for an image consisting of pure noise. It is important to note that this image can be considered as plain noise for humans, as information and meaning are truly separate concepts.

Optimization of a classifier output is performed often by comparing output probability distributions. The relative entropy, or Kullback-Leibler divergence (KL divergence) of two probability distributions is defined as

$$KL(p, q) = \sum_i p_i \log_2\left(\frac{p_i}{q_i}\right) = \sum_i p_i \log(p_i) - \sum_i p_i \log(q_i), \quad (2.8)$$

where the latter term is called the *cross-entropy* between two distributions. The cross-entropy

$$H(p, q) = - \sum_i p_i \log(q_i) \quad (2.9)$$

is a commonly used objective to be minimized in deep learning. [28] The KL divergence and cross-entropy are not symmetric metrics for distribution difference, as the measure tells how much additional information is needed for a proposed distribution q to encode data from the true distribution p . [28]

The *mutual information* between distributions measures the difference between the joint distribution $P(X, Y)$ and the distribution with assumed independence $P(X)P(Y)$:

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}, \quad (2.10)$$

where $p(x)$ and $p(y)$ are the probabilities for an event x or y independently, and the probability $p(x, y)$ is the joint probability of events x and y . The mutual information is equivalent to the difference between the entropy of a variable and the conditional entropy with the other variable: [28]

$$I(X, Y) = H(p) - H(p|q) = H(q) - H(q|p), \quad (2.11)$$

where in this case $p = P(X)$ and $q = P(Y)$. In essence, the mutual information is the reduction of uncertainty about variable X when some information is gained about Y . [28]

2.3 Unsupervised and semi-supervised learning

Unsupervised learning attempts to learn underlying structure in a dataset, using only unlabeled data during training. Unsupervised learning is not as well-defined study area as supervised learning, as the methods differ a lot and have different goals. The goal of unsupervised learning can vary from discovering new knowledge, to finding anomalies in data, or to clustering data into discrete groups. [35] Principal component analysis and knowledge mining are important subfields in unsupervised learning, but the area of interest in this thesis is clustering problems, since they relate closely to supervised learning and classification problems.

In clustering, the target is to find subgroups within a dataset so that the data points inside the groups are similar to each other while being different to the ones in other groups. As there are no labels, it can be fairly subjective to define the similarity or differences between data points. [35] Clustering is similar to classification in the way that data points are assigned to discrete groups, however semantic guidance for this grouping is lost with the labels. Sometimes if labels are available, it is possible to compare the clustering results to classification results by assigning semantic classes to each cluster and comparing the clustering groupings to the label taxonomy. [35]

If clustering is based on a measurement of distance between items in the input set, the items must be in a *metric space*. A metric space is a set where a distance can be calculated between the items of the set. For a set X and a distance function $d(\cdot)$, the following properties must hold:

- $\forall x, y \in X, d(x, y) \geq 0$,
- $\forall x, y \in X$ we have that $d(x, y) = 0$ if and only if $x = y$,
- $\forall x, y \in X, d(x, y) = d(y, x)$,
- $\forall x, y, z \in X$ we have that $d(x, z) \leq d(x, y) + d(y, z)$.

Any function that satisfies the above conditions is a metric on X . [36]

The use of metrics in the input space of images could be possible, for example by calculating the aggregated distance between pixel values. However, it is unlikely that this metric would give any information on the "true distance" between images. The perceived distance between images is usually between the semantic content of the images, not the raw pixel values themselves. A possible approach to calculating distances between images could be to learn a mapping corresponding to the semantic meaning of the image. The vector space formed by these mappings would consist of *representation vectors*, and the distance between these vectors would correspond to the semantic differences between the images. If an image of an object is augmented in some way that keeps its contents the same, the mapping to a representation vector should also stay the same. It can be said that the representation vector should be *augmentation invariant*. The unsupervised and semi-supervised methods used in this thesis use these ideas of representation vectors and augmentation invariance.

2.3.1 Invariant information clustering

Invariant information clustering (IIC) was proposed by Ji et al. [19] as a method for semantic clustering of images. IIC is based on a simple objective of maximizing mutual information between the probability vectors of an image and its transformation. Due to the conditions on metric spaces, the KL divergence equation in Equation 2.9 cannot be used as a metric between two probability vectors. Mutual information in Equation 2.10 produces a symmetric metric between two distributions, making clustering possible. The properties of mutual information, such as maximizing intra-class entropy make it a desirable distance metric for clustering. [19]

Invariant information clustering provides a conceptually simple method for clustering natural images. It is based on training a CNN that extracts features to a feature vector and then transforms this vector to C classes using a multilayer perceptron (MLP) classification head. An image x and an augmented version of the same image x' are fed to the CNN+MLP function Φ and a softmax function producing a vector $p \in [0, 1]^C$ that can be interpreted as a probability vector for the image to belong to each of the C clusters and summing to one, $\sum_{c=1}^C p_c = 1$. Because the images' contents are known to be the same, the mutual information between these distributions is the objective to be maximized [19]:

$$\max_{\Phi} I(\Phi(x), \Phi(x')). \quad (2.12)$$

The schematic for the process is also illustrated in Figure 2.1:

The optimization for mutual information maximization turns out to be useful since it avoids collapsing the clustering to a single cluster or other degenerate solutions. Due to the properties of mutual information seen in Equation 2.11 maximizing mutual information

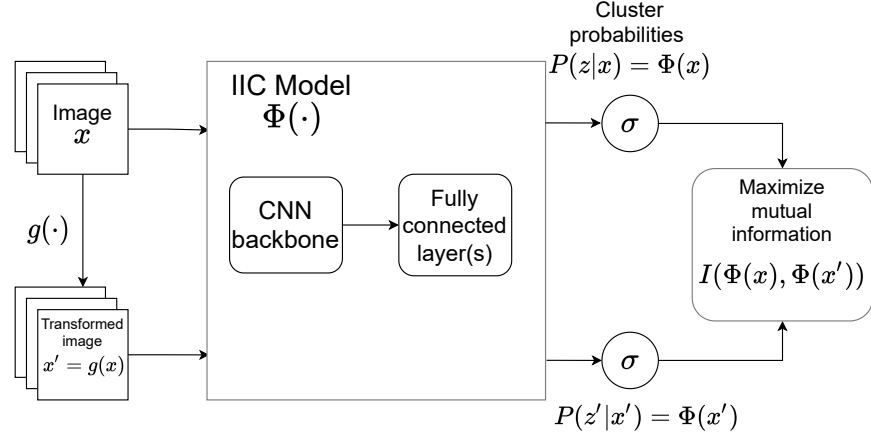


Figure 2.1. Invariant information clustering [19]

both maximizes the self-entropy of the clustering probability vector and minimizes the conditional entropy between the cluster probabilities of similar objects. The maximum value of the self-entropy of the cluster probability distribution is achieved when the distribution is uniform. This balances the assignment of images to C clusters, preventing degenerate solutions. [19]

2.3.2 Noisy student training

"Noisy student" (NS) training was proposed by Xie et al. [18] as a semi-supervised training procedure shown to improve ImageNet classification accuracy using an additional unlabeled dataset. The method is based on *knowledge distillation*, where a teacher model is used to train a student model from scratch. The overall process can be seen in Figure 2.2.

The NS training procedure is based on the noise that is added to the images during student model training. In practice this means random perturbation in the form of augmenting the images. Two datasets are used: a small dataset containing image-label pairs $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ and an large dataset $U = \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_M\}$ without labels. First, a teacher model θ^t is trained by minimizing the cross-entropy loss for labeled images

$$\text{minimize } \frac{1}{N} \sum_{i=1}^N L(y_i, f(g(x_i), \theta^t)). \quad (2.13)$$

Here $g(\cdot)$ is the noising function augmenting the training images, and $L(\cdot)$ is the cross-entropy loss function. This model is then used to generate soft pseudo-labels for the unlabeled dataset *without augmentation*:

$$\hat{y}_i = f(\hat{x}_i, \theta^t), \forall \hat{x}_i \in U. \quad (2.14)$$

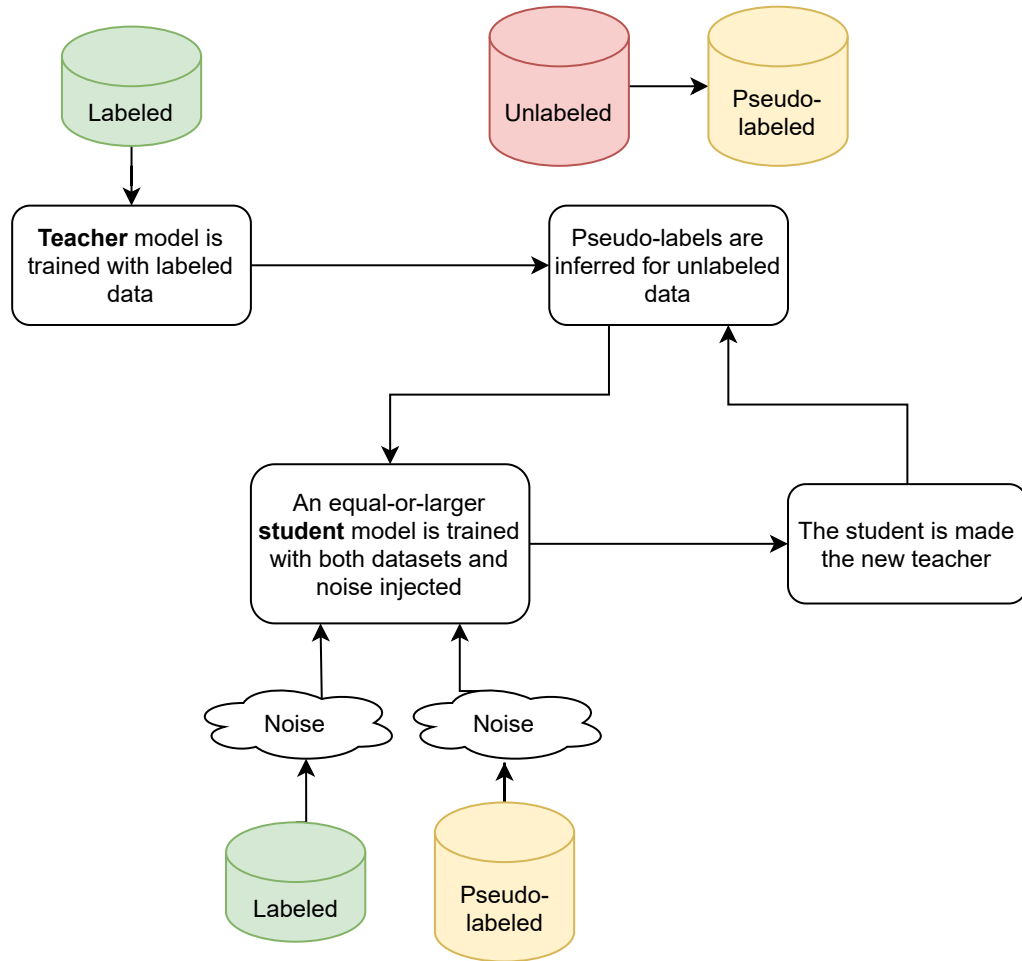


Figure 2.2. Noisy student training [18]

Finally, the student model θ^s is trained minimizing the combined loss for labeled and pseudo-labeled images:

$$\text{minimize } \frac{1}{N} \sum_i^N L(y_i, f(g(x_i), \theta^s)) + \frac{1}{M} \sum_i^M L(\hat{y}_i, f(g(\hat{x}_i), \theta^s)). \quad (2.15)$$

Xie et al. show good results on the ImageNet dataset using a large additional dataset of 300M unlabeled images. An important part of the training is finding augmentation invariant representations for the data by adding noise during the distillation process [18]. The use of augmentations during training is similar to the fully unsupervised IIC approach in Section 2.3.1.

2.4 Remote sensing

Remote sensing refers to technologies that sense the earth remotely, using for example aerial imaging or satellites. Data is often collected using higher-fidelity instruments than basic RGB-cameras. Hyperspectral [37, 38], multispectral [16], and Synthetic Aperture

Radar (SAR) [16, 39] imagery are commonly used, as they provide a larger amount of information on the earth surface than plain RGB images. A common problem in remote sensing is that clouds can cover a large area of the Earth's surface making regular, dense time series impossible. SAR fixes this problem to some extent, but the radar data is very different compared to imaging devices capturing different frequencies on the electromagnetic spectrum.

Light Detection And Ranging (LiDAR) provides additional information other imaging methods are not able to produce. Flying over the environment and scanning it with a laser beam produces 3D point clouds that show the distance to the imaging device. Combined with information on terrain height, several derivative data sources can be calculated. Canopy cover, vegetation height, and above-ground biomass are some of the values that have been collected in the forestry sciences and can be useful also in habitat mapping. [38, 40]

Probably the most common imagery source for remote sensing is multispectral satellite imagery. Satellites, such as the Sentinels by ESA and Landsats by NASA, collect daily imagery of the Earth, and the organizations operating them make the data freely and easily available. The raw data from the Sentinel satellites is processed to different levels. ESA processing can, for example, make radiometric corrections, geometric and interpolation corrections, and produce cloud masks. [41] Often these corrections are not enough for some applications and further processing is needed. Time series of data need to be collected, cloudy areas affecting analysis need to be removed, and the data needs to be stored somewhere for analysis. Geographical corrections might be needed, especially near the poles like in northern Lapland. Depending on the application, a lot of remote sensing expertise can be needed to produce usable imagery.

The Sentinel-2 satellites consist of two satellites, Sentinel-2A and Sentinel-2B, which both carry a multispectral imaging instrument with 13 spectral channels producing imagery with a 10m - 60m spatial resolution [41]. Table 2.1 shows all the spectral frequency bands where the satellites collect data and their spatial resolutions.

Remote sensing is a broad subject area with many issues to consider, such as data fusion, atmospheric physics, and different characteristics of sensors. This thesis uses highly processed and analysis-ready data, so issues that are specific to remote sensing are discussed less. The work focuses on the applications of the data and its use as input data for machine learning models.

Band number and name	S2A		S2B		Spatial resolution (m)
	Central wavelength (nm)	Bandwidth (nm)	Central wavelength (nm)	Bandwidth (nm)	
1 - Coastal aerosol	442.7	21	442.3	21	60
2 - Blue	492.4	66	492.1	66	10
3 - Green	559.8	36	559.0	36	10
4 - Red	664.6	31	665.0	31	10
5 - VRE 1	704.1	15	703.8	16	20
6 - VRE 2	740.5	15	739.1	15	20
7 - VRE 3	782.8	20	779.7	20	20
8 - NIR	832.8	106	833.0	106	10
8a - Narrow NIR	864.7	21	864.0	22	20
9 - Water vapour	945.1	20	943.2	21	60
10 - SWIR -Cirrus	1373.5	31	1376.9	30	60
11 - SWIR 2	1613.7	91	1610.4	94	20
12 - SWIR 3	2202.4	175	2185.7	185	20

Table 2.1. The Sentinel-2 bands, spectral frequencies and spatial resolutions. VRE stands for vegetation red edge, NIR for near-infrared, and SWIR for short-wave infrared. [42]

3. PRIOR WORK

Remote sensing and earth observation (EO) imagery has been used extensively for different classification tasks [12, 13, 22, 27, 43, 44, 45]. Most classification problems are related to land cover classification and segmentation, often focusing on buildings [45], farmlands [46], and roads [47]. The studies can be broadly divided to three different groups:

1. Studies focusing on remote sensing, discussing sensors, instruments, or land cover classification in general. Recently, machine learning has been widely applied to these problems.
2. Studies focusing on environment and habitat monitoring, using remote sensing as a tool for habitat mapping, classification, and monitoring.
3. Studies focusing on machine learning, where methods can be applied to remote sensing problems such as classification and segmentation.

This thesis falls jointly into the first and last groups, applying methods from machine learning literature to a general land cover classification problem, with additional contributions in sparsely annotated supervised learning. The methods are applied to habitat classification, but this work does not address the more in-depth ecological analysis beyond classification.

General land cover classification is a common problem in remote sensing [27, 44, 45, 47, 48, 49, 50, 51]. These studies apply different methods for classifying land cover to broad classes, such as built environment, agricultural areas, roads, and forests. The focus can be annotation approaches [47, 48], instrument-specific approaches [50, 51], or general classification methods [44, 45].

Earlier studies combining general land cover classification and machine learning focused on the effectiveness of different algorithms. These studies focus on per-pixel classification, where each pixel is considered as the unit of classification. Classical machine learning methods such as SVMs and random forests are common algorithms for this task. Liu et al. [44] tested support vector machines (SVMs) for land cover classification. Special focus in the paper by Liu et al. is on semi-supervised learning using SVMs. The authors use a small labeled dataset combined with a larger unlabeled one, showing a performance gain from this. Although SVMs are effective in some remote sensing tasks,

other studies have shown that random forests or CNNs can either computationally more effective, or perform better [14, 26, 52].

Random forests have shown to be highly effective in per-pixel classification problems in remote sensing. An early study by Gisalson et al. [43] uses four-channel multispectral data combined with elevation and slope information as input features to a random forest and classifies different tree species with a fairly high accuracy. Later studies have shown that random forests are still effective when training data is scarce and resolution is low, and they have become very popular in land cover classification studies [14, 26, 52, 53, 54, 55, 56].

Per-pixel classification approach is often used when the resolution of the available imagery is so low, that each pixel can be considered a single object to be classified. Object-based approaches are needed when the resolution is higher. An example of this is tree detection, where high-resolution data is used [13, 26]. The imagery used in this thesis has fairly low resolution, and each pixel can be considered a single object. However, deep learning methods can help taking the surrounding area into account.

Recently deep learning methods, such as CNNs for image or patch classification [13, 51] and fully convolutional networks (FCNs) for image segmentation [45, 47, 49, 50], have gained attention in remote sensing applications. FCNs such as the popular U-Net architecture [57] are popular in semantic segmentation, where a image or a patch is given as an input for the model and each pixel is classified to discrete classes. Kentch el al. [13] use both approaches for classifying forest types from drone-acquired imagery. The drawback of segmentation with FCNs is that the model needs a dense segmentation map (illustrated in Figure 1.3) as a ground truth annotation for training. These annotation maps are laborious to produce, and often only sparse pixel-sized annotations are available, especially in habitat classification. Another approach for modeling class distributions over larger areas is maximum entropy (Maxent) modeling. Maxent is very common in species distribution modeling [58], but is less used in habitat-related land cover classification [52, 59].

If only a minority of the pixels of an object are annotated, this can be considered *weak supervision* with *sparse annotations*. Sparse annotations could be points [27, 49, 60, 61], or scribbles [47, 48], for example. The approach proposed in this thesis can be considered weak supervision, as instead of a picture-level label the label corresponds only to the center pixel of a patch, as seen in Figure 1.3. Similar approaches can be seen in Wang et al. [27] and Laban et al. [49], where sparse point annotations are used in semantic segmentation. Wang et al. use auxiliary segmentation maps during training, which are not available in the use case of this thesis. The approach by Laban et al. uses a synthetic dataset created from the training pixel points. Because the dataset is synthetic, a segmentation map is obtained. This synthetic dataset is used as training data for a FCN

which does the classification.

Several studies have been conducted on remote sensing for environmental classification problems and applications [11, 14, 26, 52, 54, 55, 59, 62]. Petrou and Petrou [11] discuss the high-level possibilities of remote sensing in biodiversity assessment and bioindicator extraction. The authors review different areas where remote sensing has been applied to, including land cover classification for biome, ecosystem and habitat extent mapping, species distribution mapping, and detection of invasive species.

McDermid et al. [52] studied the use of remote sensing for habitat mapping, discussing different methods and strategies for gaining habitat insight from remote sensing techniques. The benefits of both unsupervised and supervised learning methods were compared, and random forests were presented as an easily interpretable machine learning technique. The challenges presented by McDermid et al. are still relevant in recent studies, for example the trade-off between per-pixel models and more spatially aware models. This thesis combines a per-pixel random forest model and a spatial CNN model to address some of these challenges.

Mäyrä et al. [26] use convolutional neural networks to detect and classify tree species from aerial hyperspectral and LiDAR data. The study shows the potential of high quality data in monitoring. Tree species classification needs high resolution aerial imagery, and additional data dimensions such as LiDAR and hyperspectral imagery provide to be useful in classification tasks. CNNs prove to be especially useful over random forests or SVMs.

Mahdavi et al. [14] review different classification methods for wetlands, a habitat class that is of high interest also in this thesis. The authors discuss different wetland classification taxonomies, but focus on remote sensing. The usefulness of multispectral data is highlighted, as infrared bands can detect moisture differences crucial in differentiating between wetland types. Elevation and LiDAR data is also brought up for detecting topographic and structural information. All these data types are also used in this thesis due to their advantages in differentiating several habitats. Wetlands are also discussed in a recent study by Magnússon et al. [55], where random forests are used to classify more fine-grained wetland vegetation types. The study focuses on temporal changes between classes and uses spatial and temporal smoothing between pixel classifications to produce more probable and accurate results.

The Natura2000 habitat types are of special interest in Europe due to the taxonomy's relation to legislation. It is thus a common target in habitat classification in several studies [53, 59, 63, 64]. Stenzel et al. [59] use maximum entropy modeling to detect Natura2000 grassland and wetland classes from 5m resolution EO imagery. Alkaline fens, another Natura2000 wetland class, is studied by Kopel et al. [53] in a study where random forests are used for this single-class classification task.

Studies focusing on machine learning and deep learning tend to deal with fairly general concepts that can be applied to almost any classification or segmentation task. Unsupervised and semi-supervised learning has gained lots of attention lately, with several papers published only in the year 2020 proposing approaches to the problem [65, 66, 67, 68, 69, 70]. The common denominator in these papers is using data augmentation to produce several versions of the same image and using the knowledge that the derivations represent the same object as the key for learning. The actual implementations are very different and rely on different loss function choices and architectural tricks.

4. PROPOSED METHOD

This thesis compares different machine learning models for the classification of finely-grained habitat classes using remotely sensed imagery as input data. The focus is on testing different training approaches related to training a CNN classifier on data that has the following characteristics:

1. Annotated ground truth data is scarce
2. There are large amounts of unannotated data available
3. Annotations are available for single pixels only
4. Target taxonomy is fine-grained and the distribution among classes is highly imbalanced

One of the objectives was to evaluate the feasibility of using CNNs compared to traditional pixel-based methods. The hypothesis was that the pixel surroundings might contain some information that is useful in classification, making a CNN suitable for this kind of problem. The CNN should be evaluated against common machine learning methods used in remote sensing, such as random forest classifiers. Random forest input is usually the channel values of a single pixel and the output is the class for this pixel. The CNN approach would similarly classify only a single pixel, but use the whole neighborhood of that pixel as the input. Usually image classification gives an output label on the whole image level, so the semantic content can be anywhere in the image. Here, the semantic content is only in the center pixel, but the neighborhood can contain useful information for determining the class for that pixel. Figure 1.3 illustrates this difference.

The CNN approach differs also from image segmentation, which classifies all of the pixels in the image that is given as an input. The third characteristic of the above listing prevents using a traditional image segmentation approach, since annotations are very sparsely available. Some experiments were done by training a classifier with single-pixel labels, classifying all unannotated pixels and using these maps as ground truth data, but the approach did not perform as well as by just classifying each pixel separately.

The first and second characteristic make the case for testing unsupervised and semi-supervised learning approaches. Since there are large amounts of unannotated data available, maybe it can be utilized somehow? The recent advances in unsupervised and

semi-supervised learning on benchmark datasets makes it possible to apply the findings to a new dataset like this. The hypothesis for using semi-supervised learning is that a model trained initially with only a smaller dataset can learn more general features and perform better if training is continued in a semi-supervised fashion. The methods chosen to be tested in this thesis are Invariant Information Clustering (IIC) proposed by Ji et al. [19] and Noisy Student training proposed by Xie et al. [18] due to their conceptual simplicity and promising results on benchmark datasets. These approaches are discussed in Section 2.3.

Another way of tackling the problem of a small dataset could be using *transfer learning*. This is a common practice in machine learning, where a model is pretrained with a larger dataset and fine-tuned to a specific task. The benefits of transfer learning compared to random initialization of weights are well-known [71]. Pretraining the model is often done with large RGB datasets of natural images, such as the ImageNet or COCO datasets. In a remote sensing setting pretrained models are harder to find. Sentinel-2 datasets are available, but in the case of this thesis the data is unique due to the used laser and phenology rasters. However, because the source rasters are large and land cover annotations can be found in the form of CORINE land cover classifications, we show that these out-of-domain classes can be used successfully for pretraining followed by transfer learning to a finer class taxonomy.

Pretraining with a larger dataset is common procedure in applied deep learning. Often remote sensing applications can have a very specific classification taxonomy, where data collection could be expensive, leading to a small dataset to train a model on. A larger dataset of the same domain might be available, such as the CORINE land-cover labels that cover most of Europe. Although the CORINE-classes are quite different from the final classes, pretraining a CNN with a large CORINE-dataset improves classification performance significantly. Details of the CORINE dataset are discussed in Chapter 5.

The proposed workflow for training a model can be seen in Figure 4.1. The model is first trained with a large dataset with CORINE land cover class labels or alternatively in a fully unsupervised manner using IIC training. Then, the small dataset is used to fine-tune the model to either the Natura2000 or GCS target classes. Finally, training of the fine-tuned model is continued by applying Noisy Student semi-supervised learning and training a new model with the help of a larger dataset. Because the Noisy Student is essentially a distillation training method, all layers of the new model are trained.

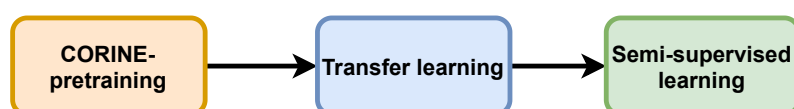


Figure 4.1. The proposed training outline for classification

A special characteristic of the dataset is that the target taxonomies are finely-grained and highly imbalanced. To address this, data augmentation and ensembling with other approaches is needed to produce more generalized and robust models. This thesis proposes using a special kind of augmentation, center-fixed cropping, to make the model focus more on the center pixel that is the one being classified. Pixels nearer the center should have higher importance on the classification results than ones further away. This could be achieved with different ways. However, approaches by weighting the pixel values differently did not work as well as just randomly cropping the image by keeping the center pixel in the same position. The cropping size is chosen randomly for each batch of images separately and all the images in a single batch have the same dimensions. These augmentations are used also during inference, when *test-time augmentation* (TTA) is applied. A pixel is classified several times with different augmentations applied. The final classification is the average of several augmented classifications.

A novel approach in this thesis is addressing all of the special properties by adding a random forest classifier to the center pixel only and ensembling it with the CNN model. The CNN model is trained in a semi-supervised manner, using large amounts of data and tackling the small dataset problem. Because annotations are available only for single pixels, the focus should be on them. Classifying single pixels both with a CNN and the random forest, shown to be effective in remote sensing [43, 54, 59], combines the best sides of both models. CNNs are great with spatial awareness and random forests excel on imbalanced data [72, 73]. An ensemble model performs better than either model alone, as is shown later in Chapter 7. The overview of the inference process on an image, using the ensemble model is shown in Figure 4.2.

Sometimes inducing biases for classification by choosing a taxonomy is not desired. Field-collected ground truth classes could be impossible to distinguish from the remote sensing data, making a classification problem with a taxonomy like this an ill-posed one. An alternative approach is to produce fully unsupervised segmentations and use them for analysing the environment and the available data. This approach is tested by training some models fully unsupervised with the IIC approach. The validity of these models is evaluated qualitatively, but they provide good insights on how a fully unsupervised method can be used when training data is abundant but annotations are scarce. These segmentation maps find a pre-defined number of C maximally different clusters from the training dataset, making it easier to understand what kind of features can be distinguished from each other from the source data. Deep learning and CNN based approach make it possible to learn complex pattern features instead of relying to pixel-based unsupervised methods.

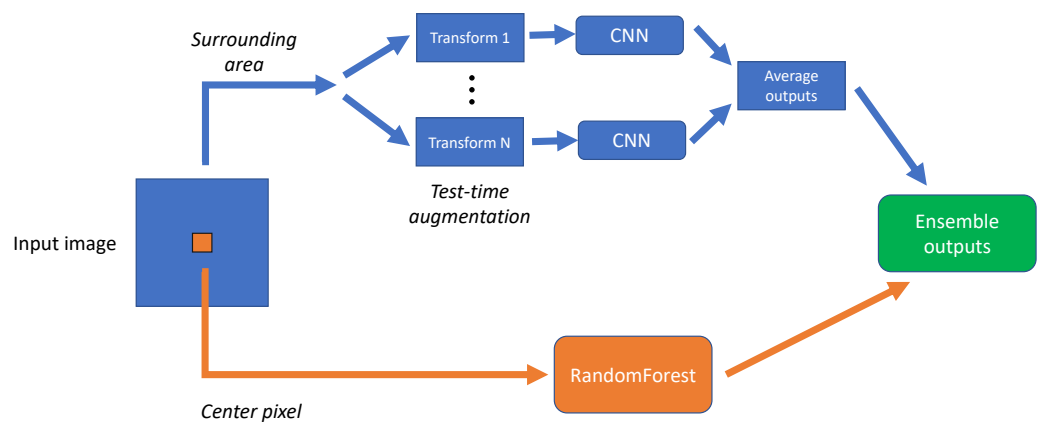


Figure 4.2. Classifying an image using the CNN+random forests ensemble hybrid model, with test-time augmentation

5. DATA

The georeferenced point annotation dataset used in this work is not currently publicly available, so the details are discussed in this chapter. The differences between classification taxonomies are discussed, and the Finnish habitat names used throughout this thesis are translated for reference. The final remote sensing data is also defined in this chapter. The technical details of processing the data used in the experiments is discussed later in Chapter 6.

5.1 Annotations

The point annotations used in this thesis were collected by Metsähallitus during summer 2020. The dataset consists of 2558 georeferenced points representing the habitat information within a 10m radius of the annotated point. The dataset contains rich information about the sample site, including plant species information, vegetation height and cover approximations, and degradation information, for example. Most of these attributes could be used as a target for machine learning purposes, but this thesis focuses on the two classification taxonomies present in the dataset: the Natura2000 classes [74] and the General Classification System for Finland's biotopes (GCS) [75] classes.

The Natura2000 classification taxonomy defines different European natural habitat types and is widely used across Europe in habitat conservation. The taxonomy lists 233 different habitat types, including 71 priority classes, which are in danger of disappearance. The Natura2000 classification system has been designed for nature conservation, with recognition of endangered habitats in mind. [74] The General Classification System for Finland's biotopes is similarly a classification taxonomy for biotopes, produced by the Finnish Environment Institute and Metsähallitus, a state-owned environment services provider. The General Classification System is used as a tool for categorizing biotopes in a way that is suitable for recognition from aerial photographs and mapping data. The GCS classes are chosen so that they produce large, over 2500 m² areas, making it easier to distinguish them from remote sensed data.

The two classification taxonomies both represent biotope and habitat classes, but the focus differs between them. The Natura2000 classes in this dataset are more high-level and represent larger entities, while the GCS classes are fairly detailed down to different

Natura code	Finnish name	English name [74]	Count
3110	Karut kirkasvetiset järvet	Oligotrophic waters containing very few minerals of sandy plains (Littorelletalia uniflorae)	28
3160	Humuspitoiset järvet ja lammet	Natural dystrophic lakes and ponds	1
3220	Tunturijoet ja purot	Alpine rivers and the herbaceous vegetation along their banks	4
4060	Tunturikankaat	Alpine and Boreal heaths	472
4080	Tunturipajukot	Sub-Arctic Salix spp. scrub	9
6150	Karut tunturiniityt	Siliceous alpine and boreal grasslands	121
6270	Runsaslajiset kuivat ja tuoreet niityt	Fennoscandian lowland species-rich dry to mesic grasslands	1
6430	Kosteat suurruohoniityt	Hydrophilous tall herb fringe communities of plains and of the montane to alpine levels	13
6450	Tulvaniityt	Northern boreal alluvial meadows	46
7140	Vaihettumissuot ja ranta-suot	Transition mires and quaking bogs	165
7160	Lähteet ja lähdesuot	Fennoscandian mineral-rich springs and springfens	104
7220	Huurresammallähteet	Petrifying springs with tufa formation (Cratoneurion)	8
7230	Letot	Alkaline fens	26
7240	Tuntureiden rehevät puronvarsisuot	Alpine pioneer formations of the Caricion bicoloris-atrofuscae	2
7310	Aapasuot	Aapa mires	27
7320	Palsasuot	Palsa mires	17
8110	Tuntureiden vyörysoiraikot ja -lohkareiko	Siliceous scree of the montane to snow levels (Androsacetalia alpinae and Galeopsietalia ladani)	7
8210	Kalkkikalliot	Calcareous rocky slopes with chasmophytic vegetation	2
8220	Silikaattikalliot	Siliceous rocky slopes with chasmophytic vegetation	64
9010	Luonnonmetsät	Western Taïga	271
9040	Tunturikoivikot	Nordic subalpine/subarctic forests with Betula pubescens ssp. czerepanovii	453
9050	Lehdot	Fennoscandian herb-rich forests with Picea abies	58
9080	Metsäluhdat	Fennoscandian deciduous swamp woods	12
91D0	Puustoiset suot	Bog woodland	19
91E0	Tulvametsät	Alluvial forests with Alnus glutinosa and Fraxinus excelsior (Alno-Padion, Alnion incanae, Salicion albae)	106
Total			2036

Table 5.1. Natura2000 classes with their Finnish names used in this thesis, along with their English name according to [74], and the number of samples in the full dataset

dryness levels of the environment. Both taxonomies are hierarchical and have a unique identifier for each class. For example the Natura2000 '7310 Aapa mires' class belongs to the '73XX Boreal mires' group, which belongs to the highest '7XXX Raised bogs and mires and fens' group. The GCS classes have a similar hierarchy on three levels.

The annotated dataset of 2558 points does not contain Natura2000 or GCS information for each point. The full dataset contains 25 unique Natura2000 classes and 36 unique GCS classes. The amount of samples from these classes is highly imbalanced, with the smallest classes containing only one sample, but the largest ones containing 472 and 396. The classes present in the dataset are listed in Tables 5.1 and 5.2. The tables also list the number of samples in each class.

The field work for collecting accurate, georeferenced information in a remote location is

GCS code	Finnish name	English translation	Count
101	Kalliolaet, -rinteet ja terassit	Rocky summits, slopes, and terraces	26
102	Kalliojyrkänteet ja seinämät	Rocky cliffs and cliff faces	16
103	Kalliorotkot	Rocky gorges	2
104	Louhikot ja kivikot	Block fields	66
105	Vyörylouhikot ja -kivikot	Siliceous scree of the montane	6
220	Kasviton kivennäismaa	Vegetation-free mineral soil	39
231	Jäkälä (karukkokangas)	Lichen (oligotrophic)	5
232	Jäkälä-varpu (kuiva)	Lichen-dwarf-shurb (dry)	115
241	Jäkälä-sammal-varpu (kuivahko)	Lichen-moss-dwarf-shrub (dryish)	306
242	Sammal-varpu (tuore)	Moss-dwarf-shrub (mesic)	396
251	Sammal-varpu-ruoho (lehtomainen)	Moss-dwarf-shrub-grass (herb-rich forest-like)	145
252	Ruoho (lehto)	Grass (herb-rich forest)	63
261	Jäkäläinen heinä-sara	Poaceae and Carex with lichen	3
262	Sammaleinen heinä-sara	Poaceae and Carex with moss	93
263	Ruohoinen heinä-sara	Poaceae and Carex with grass	136
271	Tuntureiden sammalpinnat	Moss covered fells	120
311	Varsinaiset korpisuot	Actual wooded minerotrophic mires	9
312	Korpi-välipintasuot	Wooded minerotrophic - lawn mires	14
313	Korpi-rimpipintasuot	Wooded minerotrophic wet mires	20
321	Varsinaiset rämesuot	Actual dwarf-shrub mires	63
322	Räme-välipintasuot	Dwarf-shrub lawn mires	14
323	Räme-rimpipintasuot	Dwarf-shrub wet mires	48
324	Räme-vesipintasuot	Dwarf-shrub flooded mires	1
331	Välipintasuot	Lawn mires	13
332	Väli-rimpipintasuot	Wet lawn mires	26
333	Rimpipintasuot	Wet mires	33
334	Vesipintasuot	Flooded mires	1
335	Arokosteikot	Grassy wetlands	5
336	Tihkupinta	Seepage wetlands	37
410	Avolähde	Open spring	31
422	Puro (leveys <2 m)	Stream (width <2 m)	4
424	Leveä joki (>5 m)	Wide river (width >5 m)	1
430	Järvi tai lampi	Lake or pond	38
522	Tuore niitty	Fresh meadow	0
523	Kostea niitty	Wet meadow	1
645	Poroerotuspaikat	Reindeer herding site	5
Total			1901

Table 5.2. General Classification System classes with their Finnish names used in this thesis, along with an English translation by the author, and the number of samples in the full dataset

demanding and expensive. Because of this, the 2558 point dataset is fairly large in this domain, but still remains small for deep learning purposes. To combat this challenge, an additional dataset was collected using the CORINE land cover classification of Finland [76, 77]. 35 600 points were randomly sampled from northern Lapland with minimum distance of 500m between points. The sampling was bounded by the area shown in Figure 1.2. The class value of the CORINE land cover raster pixel under the chosen point was set as the annotation for each point. The CORINE land cover includes 44 classes

in five major groups: artificial surfaces, agricultural areas, forests and semi-natural areas, wetlands, and water bodies. The class taxonomy is very broad, covering most general land-cover classes. The General Classification System and Natura taxonomies are more finely grained, with large number of very similar classes, for example in the wetlands.

5.2 Remote sensing data

Producing useful data products from raw remote sensing data requires a lot of expertise. This work uses several high-quality data products produced by the Finnish Environment Institute remote sensing experts:

- Processed laser scanning data for canopy cover and vegetation height,
- Processed Sentinel-2 imagery for 9 selected bands (2,3,4,5,6,7,8,11,12),
- Sentinel-2 NDVI phenology data.

The first dataset contains laser point clouds processed to 8m spatial resolution rasters. Two rasters are used, first containing the mean height of the vegetation in the pixel area, and the second containing the percentage of canopy cover in the pixel area. The Sentinel-2 imagery dataset contains a selection of 9 of the available bands, mosaiced from imagery collected in summer 2020. The 60m spatial resolution bands are left out, as well as the 8A band highly correlating with band 8. The laser scanning and NDVI phenology data is not currently publicly available.

The normalized difference vegetation index (NDVI) is a well-known index in remote sensing that indicates the amount of green vegetation in an area. The index is calculated as the ratio between the difference and sum of the NIR and red channels of a multispectral instrument:

$$NDVI = \frac{NIR - RED}{NIR + RED}. \quad (5.1)$$

The NDVI is a good indication of the vegetation in the area. This information can be used to calculate phenological information or the yearly change in the environment. The phenology rasters contain the sum, maximum and amplitude of the NDVI index for the year 2020.

A part of this thesis was to make the rasters usable for machine learning purposes. This means unifying the spatial resolutions and combining the bands together to produce a separate dataset raster. All rasters not in the 10m spatial resolution were resampled to match the 10m resolution using nearest-neighbor sampling. Because the range of the values between rasters changes depending on the source, the rasters need to be standardized before model training. For this purpose, the mean and standard deviations for each raster were calculated and used during training to standardize each channel/band

of the training image.

The source rasters provide a good overview of the environment and highlight different features from it. Good data sources are in a way "orthogonal" to each other, i.e., each channel tells something about the observed environment that the other channels do not. Figures A.1 and A.2 in the appendix illustrate all of the source data rasters from the areas used to illustrate classification results in Chapter 7. It can be seen that some of the Sentinel-2 channels are highly correlated, giving less additional information, but the phenology channels, for example, provide aggregated time-series information that is highly useful in vegetation classification.

6. EXPERIMENTAL SETUP

This chapter deals with the technical details of the experiments. The deep learning architectures, hyperparameters, and training details are specified, along with the random forest parameters. Used tools, frameworks, and software packages are specified. Technical details about producing the training data is also discussed.

6.1 Data

The annotation data and the source imagery described in Chapter 5 could not be used by themselves. After resampling the rasters to the same spatial resolution of 10m, a combination raster was created, consisting of 14 bands: Canopy cover, vegetation height, NDVI amplitude, NDVI sum, NDVI maximum, and Sentinel-2 channels 2, 3, 4, 5, 6, 7, 8, 11, and 12.

This full raster spanning the entire northern Lapland area in 10m resolution is over 70Gb in size. It cannot be used directly as an input for machine learning models, and loading this raster and accessing it during training would be unfeasible. Because of this, a separate dataset was sampled from this raster. 100m radius areas around the annotation points were collected, producing in essence an image dataset consisting of 19x19 images with 14 channels, with an corresponding annotation. Similar dataset was collected for the 35 600 randomly sampled images. CORINE labels from the CORINE land cover dataset [77] were added as annotations for each image.

A problem with these 19x19 pixel images is that they overlap each other, making splitting the data for training and testing challenging. The datasets were further processed by choosing a test set of about 20% of available annotations for each taxonomy, in a stratified manner, and removing all the other images that geographically overlap or touch this test set. The remaining non-overlapping images were used as the training data. This was done five times to produce five cross-validation folds, with test sets not containing common images. Figure 6.1 illustrates this train-test-split.

Some problems and data leakage can arise due to the close proximity of train and test samples. A way to prevent this would be to geographically choose the training and testing areas. However, some sample classes are bound to a small area, and a geographical

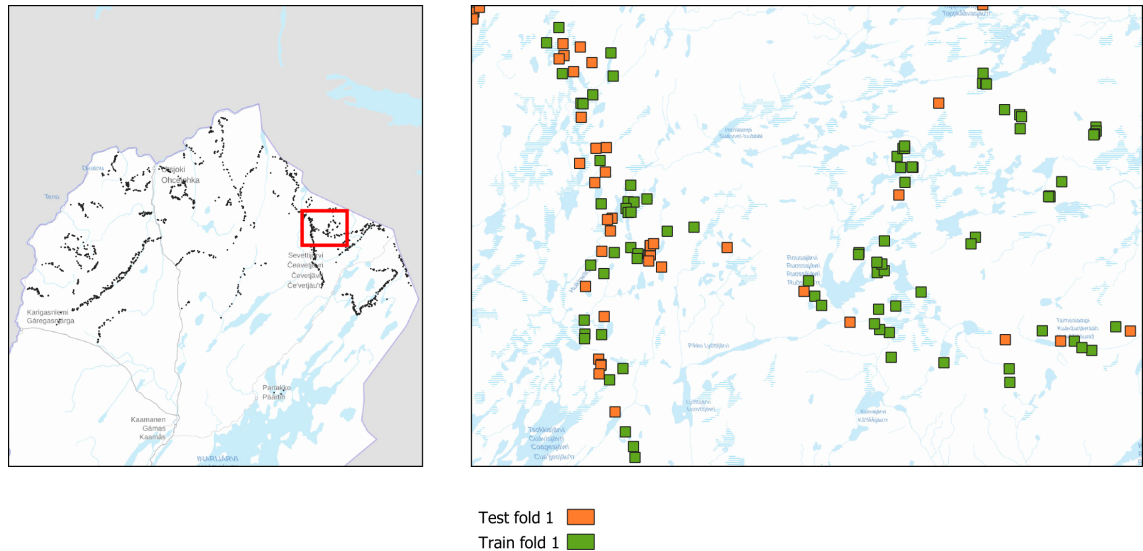


Figure 6.1. Example of the first cross-validation fold train-test-split. The data is split so that training and test sets do not overlap. Close geographical proximity can however produce some data leakage due to distinct features like rivers and fells.

split between train and test sets would rule a lot of classes out from the analysis.

6.2 Machine learning setup

For the CNN, a ResNet18 architecture [33] was used as a backbone. The first convolutional layer was changed to match the amount of channels in the image and the final fully connected layer was changed to match the number of classes. Experiments were done also with ResNet50 but they yielded similar results as the ResNet18, so ResNet18 was chosen as the final model due to less parameters and faster training. A fully connected classification head is used for projecting the 512-dimensional feature vector into C class predictions.

All of the models are trained with same hyperparameters and optimizers to ensure comparability. Optimization is done using the Adam algorithm [78] with a learning rate of $1e-4$. Batch size of 128 was chosen for all tests. Series of test were conducted to choose training time of 500 epochs for the transfer learning and semi-supervised learning tasks, with pretraining of 50 epochs for the CORINE pretrained models.

Tests on different augmentations were also conducted. The results presented in this thesis used simple random horizontal flip augmentations and a Gaussian blur augmentation which is applied to all channels. The random cropping augmentation described in Chapter 4 was used for part of the experiments.

The unsupervised IIC models were trained for 15 epochs, for both classification heads, totalling 30 epochs for each model. The 19×19 IIC model was used as the feature extrac-

tor for transfer learning models using unsupervised pretraining as the feature extractor. For fully unsupervised segmentation, different window sizes and cluster amounts were used. Models were trained with window sizes of 3x3, 9x9, and 19x19 and with 10, 30, and 70 clusters.

The random forest classifier was used both for comparison and as part of the ensemble classifier. A decision forest consisting of 100 decision trees trained with Gini impurity splitting and no restriction on tree depth was used. Grid parameter search was conducted for different forest sizes, maximum tree depths, and node impurity measures. The best parameters of this search indicated that a forest size of 900 trees would be the best one by absolute measures, but the speed-accuracy trade-off was better with the 100 tree forest.

All programming was done using Python as a programming language, the Pytorch library [79] for deep learning and scikit-learn library [80] for general machine learning and preprocessing. Processing the geospatial data and rasters were done with the open source QGIS application [81]. Final classification of the full Lapland area was parallelized using the Dask library [82]. Experiments were tracked using the Weights and Biases - experiment tracking service [83]. Computation was done on CSC (IT Center for Science, Finland) computing clusters.

6.3 Evaluation metrics

The results are evaluated using commonly used evaluation metrics for classification. Top 1,3, and 5 accuracies, f1-score, precision and recall are calculated using their standard definitions. Threshold metrics such as average precision (AP) and area under the return-on-characteristics curve (ROC-AUC) are also calculated.

When dealing with imbalanced data in multiclass classification, the averaging method across classes is important. In order to get the most comprehensive picture of the classifier's performance, both macro-average and weighted average across classes is calculated for each metric. Macro-average calculates a metric (f1, precision, recall, AP, AUC) for each class and outputs a unweighted mean across classes. Its counterpart, the micro-average, aggregates the predictions of all categories and calculates the metric as if the classification were binary. Weighted average is similar to macro-averaging, but each class is weighted by the amount of samples in it. Micro-averaging is generally not useful in multi-class classification since precision, recall, and accuracy get the same values [84]. The macro-average is biased towards smaller classes, making weighted average a good compromise. The downside of weighted averaging is that the averaged f1-score is not necessarily the harmonic mean of precision and recall. A special characteristic of weighted recall is that it is the same value as accuracy.

When performing cross-validation, metrics are calculated for each fold separately. These

need to be averaged. A common approach in machine learning is to just (macro) average the results across folds. It has been noted that this can produce slight bias to the results if the folds are not stratified [85], making micro-averaging across cross-validation folds a better, yet more complex option. The folds in this thesis are stratified and similarly sized, so the more traditional arithmetic mean across folds is used.

A classifier outputs a probability value between 0 and 1 for each class. In binary classification, the choice of a classification threshold leads to a tradeoff between sensitivity (true positive rate, TPR) and specificity (false positive rate, FPR). Similar tradeoff happens between precision and recall: when recall increases, precision usually decreases. The relationships between these metrics can be calculated for each threshold using return-on-characteristics (ROC) and precision-recall curves. The former plots classifier sensitivity against specificity, and the latter plots precision against recall for each threshold value. For multiclass classifiers, the thresholding is done by considering each class separately against all other classes. For each threshold, averaging can be performed by micro or macro-averaging. Again, micro-averaging is biased towards the most common classes, while macro-averaging gives more weight to smaller classes. The results show both curves to illustrate the performance across classifiers better.

7. RESULTS

This chapter shows the final results of the experiments. The results are roughly divided to classification results and classification maps. The models were evaluated using several metrics and cross-validation. Selected metrics are displayed in this chapter for brevity, while the full results with standard deviations can be found from the appendix. The results are assessed with ROC-AUC curves, precision-recall curves and confusion matrices. Results for higher hierarchies are also discussed

Sensitivity studies display the effects of different training choices on the models. The effects of these choices are compared for the best performing model as well as for aggregates of several models. All metrics and sensitivity studies are calculated for both class taxonomies, Natura2000 and GCS.

The classification maps show the final classification maps produced by the models. Different models and their results are compared for both class taxonomies. Model outputs, or confidence maps, are also displayed and discussed. Finally, fully unsupervised segmentation maps are reviewed.

7.1 Classification

For classification, in total 11 different CNN models, shown in Table 7.1, were trained with five-fold cross-validation. The effect of pretraining and semi-supervised learning on classification performance was tested. In addition, the effect of three alternative training approaches were tested: random crop augmentation proposed in Chapter 4, convolutional layer freezing, and test-time augmentation (TTA). Test-time augmentation consistently improves results and all tables in this chapter are presented with TTA applied, unless stated otherwise.

An overview of the 11 different CNN models can be seen in Figure 7.1. Pretrained (PT) models use a pretrained CORINE-model, either fine-tuning also the convolutional layers or freezing them. Base model without any pretraining (PT) was trained as a baseline, and a model pretrained in a fully unsupervised manner model (UPT) was also tested. The pretrained models are then used as a teacher model for Noisy Student semi-supervised learning.

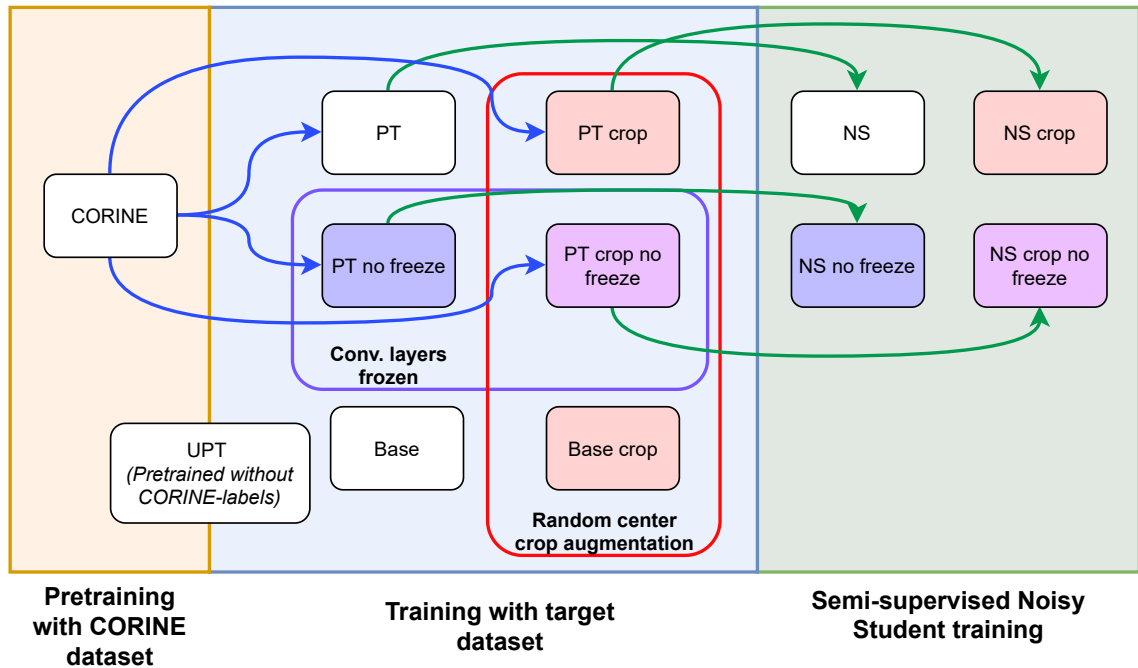


Figure 7.1. Relationships between different trained models. The Base models are trained only on the small target dataset. PT models are transfer learned from a model trained on a large CORINE-labeled dataset. NS models are a continuation of the PT models, using the PT models as a teacher model during training. The UPT model is pretrained unsupervised using the CORINE dataset without labels and fine-tuned using the target dataset

The appendix contains full results for both classification taxonomies in Tables B.2 for GCS classes and B.7 for Natura2000. The tables show the cross-validated results for several metrics, as well as the metrics' standard deviations. The same metrics for the random forest model are shown in Tables B.1 for GCS classes and B.6 for Natura2000 classes. The full results for the ensemble model are similarly in Tables B.3 for GCS and B.8 for Natura2000.

An overview of the full results is given in Tables 7.1 and 7.2 for Natura2000 classes and in Tables B.4 and B.5 in the appendix for the GCS classes. The tables show the results for ResNet models by themselves and ensembled with the random forest model. Overall observations from the results are:

- Natura2000 classes are significantly easier to classify than GCS classes,
- Random forest beats all plain ResNet models,
- Ensemble models perform better than both random forests and ResNets,
- Models that train only the classification head perform better than non-frozen or semi-supervised models,
- Random center cropping improves performance in some cases.
- Test-time augmentation consistently improves evaluation results.

metric model	F1 weighted	Prec. weighted	Rec. weighted/Acc	Top3 acc
Base	0.481	0.469	0.509	0.751
Base crop	0.491	0.479	0.536	0.791
NS	0.502	0.500	0.541	0.795
NS crop	0.507	0.505	0.554	0.800
NS crop no freeze	0.453	0.456	0.515	0.748
NS no freeze	0.493	0.478	0.526	0.780
PT	0.514	0.517	0.550	0.794
PT crop	0.493	0.495	0.554	0.810
PT crop no freeze	0.504	0.498	0.537	0.784
PT no freeze	0.486	0.475	0.508	0.775
UPT	0.366	0.321	0.439	0.713
Random forest	0.539	0.533	0.579	0.813

Table 7.1. ResNet results for Natura2000 classes: Selected metrics for Natura2000 classification for different ResNet models and the random forest baseline, with test-time augmentation. Trained models include a baseline model (Base) trained only with the final small dataset, a CORINE-pretrained model (PT), model pretrained in an unsupervised manner (UPT), and the PT model continued with semi-supervised Noisy Student learning (NS). Some models have alternative models with cropping or convolutional layer freezing applied.

Overall best performance for both taxonomies were achieved with a plain transfer learning approach from the CORINE-pretrained model (PT in the tables), without cropping augmentations or further semi-supervised learning. The overall performance of the baseline random forest model is better than any of the ResNet models alone, but ensembling these models together boosts performance significantly. All of the CNN approaches are very close to each other, so the possibility of noise should be taken to account, especially since the variance between cross-validation folds is high. Test-time augmentation (TTA) produces consistently better results, and these tables illustrate only the TTA results with 5 augmentations. The appendix contains full comparison between TTA and non-augmented results and the difference is further illustrated in section 7.1.1.

The performance across different classes can be seen in the precision-recall and ROC curves in Figures 7.2 and 7.3. The performance difference between classes is substantial, with more populous classes being more reliably classified. This can be seen also in the difference between the macro- and micro-averages of the classifiers over classes, drawn in bold. Due to the larger classes containing most examples, and them being classified mostly correctly, the micro-average performance is higher. Because the dataset contains

metric model	F1 weighted	Prec. weighted	Rec. weighted/Acc	Top3 acc
Base	0.494	0.478	0.527	0.805
Base crop	0.504	0.494	0.553	0.823
NS	0.519	0.512	0.565	0.820
NS crop	0.534	0.525	0.586	0.822
NS crop no freeze	0.522	0.521	0.585	0.812
NS no freeze	0.517	0.504	0.555	0.816
PT	0.543	0.550	0.590	0.820
PT crop	0.532	0.545	0.589	0.831
PT crop no freeze	0.512	0.511	0.551	0.823
PT no freeze	0.499	0.485	0.526	0.823
UPT	0.484	0.484	0.563	0.794
Random forest	0.539	0.533	0.579	0.813

Table 7.2. Ensemble results for Natura2000 classes: The models of Table 7.1 ensemble with the random forest model. Refer to Table 7.1 for abbreviations

several classes with only few examples and the classifier performing poorly on these, the macro-averaged classification performance is considerably lower.

Figure 7.4 illustrates the differences between different models and the effect of test-time augmentation (Similar figure for GCS classes in Appendix C.2. As Figures 7.4a and 7.4b show, the ensemble model performance gain is higher for the macro-averaged models, indicating better performance in smaller classes. The ensemble model gain is smaller in the micro-averaged models, due to better random forest performance in the larger classes.

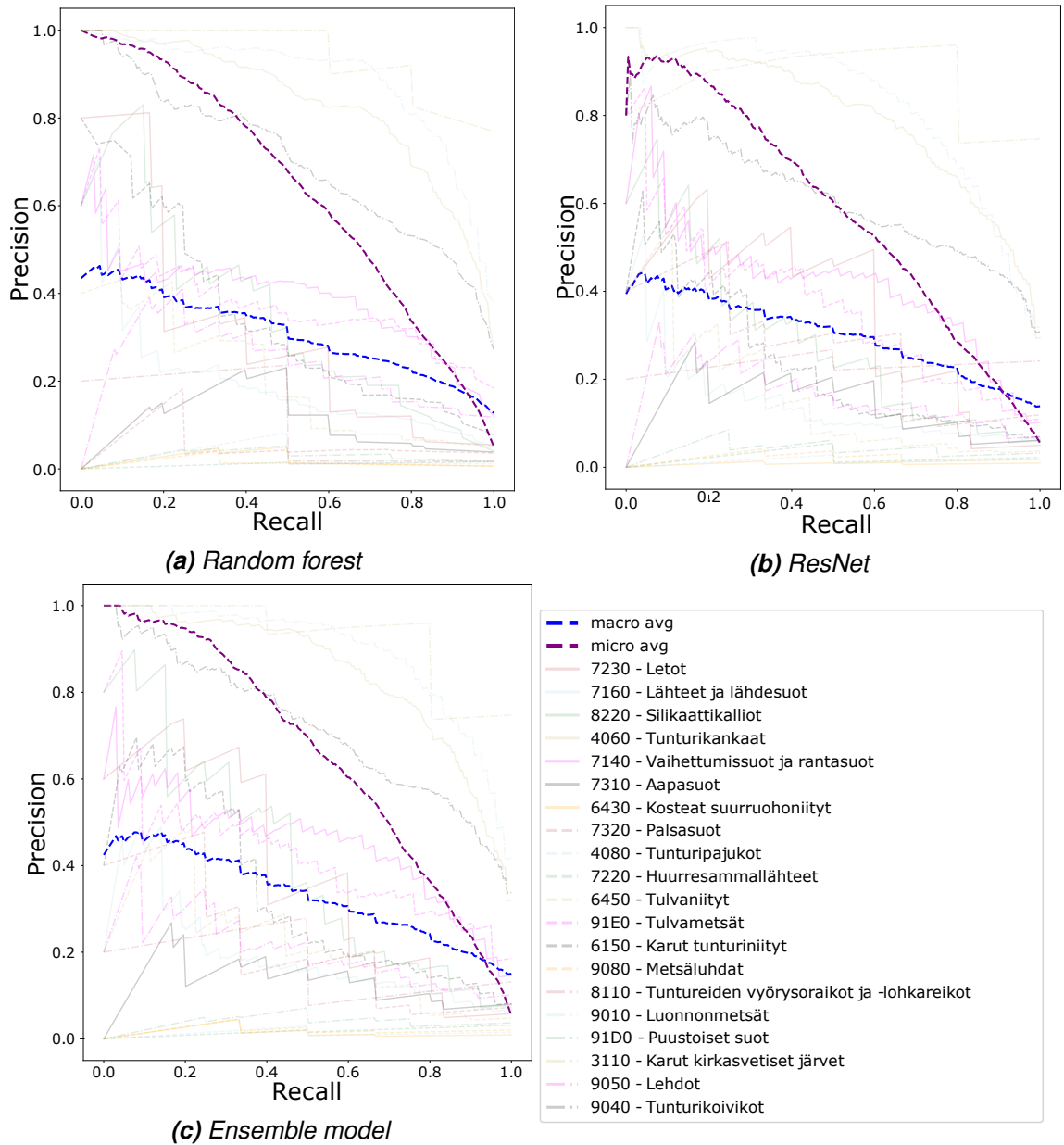


Figure 7.2. Class-wise precision-recall curves for Natura2000 classifications using the CORINE-pretrained model and test-time augmentation

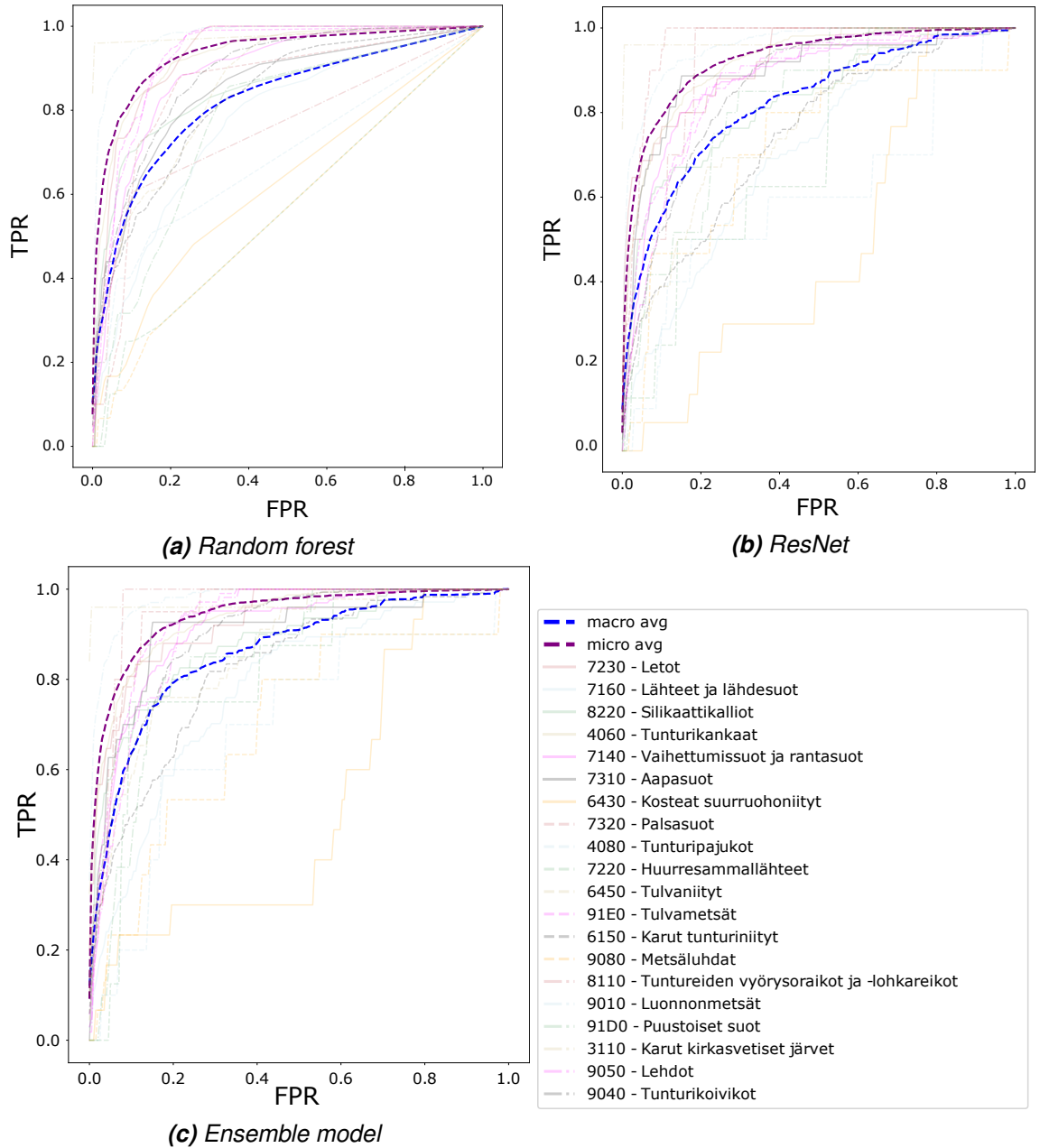
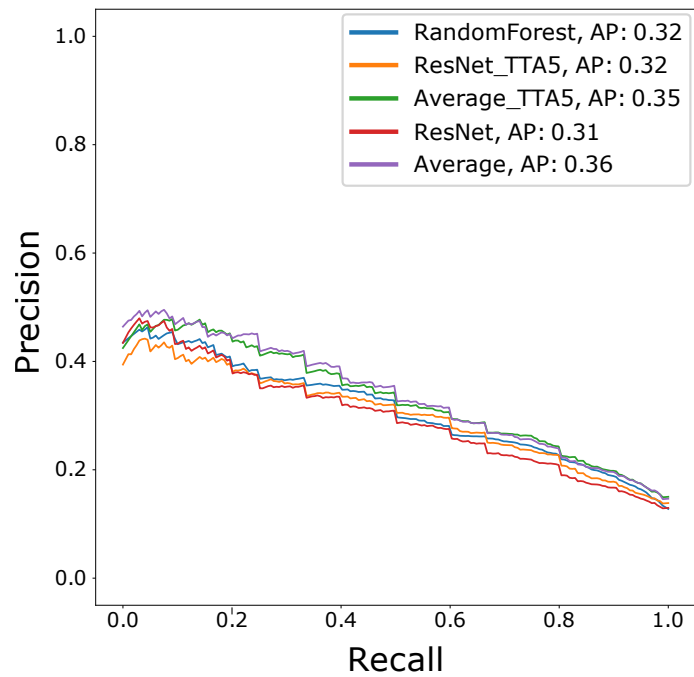
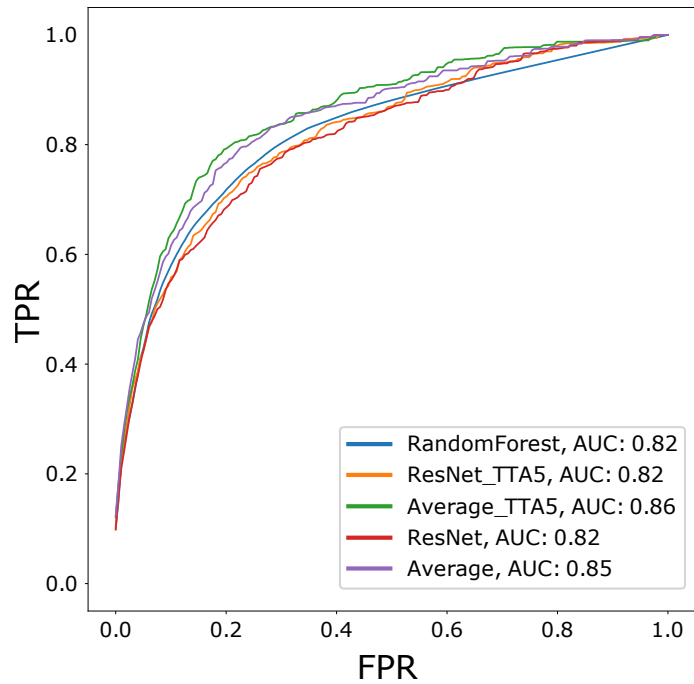


Figure 7.3. Class-wise ROC curves for Natura2000 classifications using the CORINE-pretrained model and test-time augmentation



(a) Macro average precision-recall curves

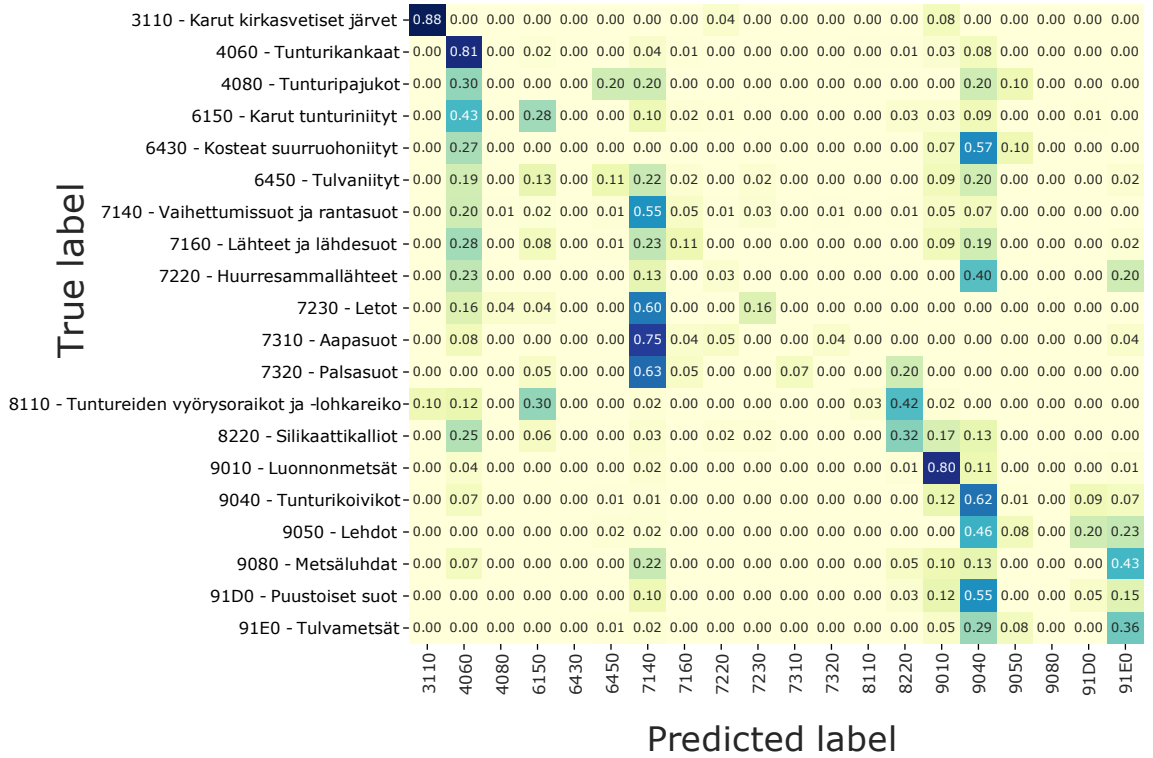


(b) Macro average ROC curves

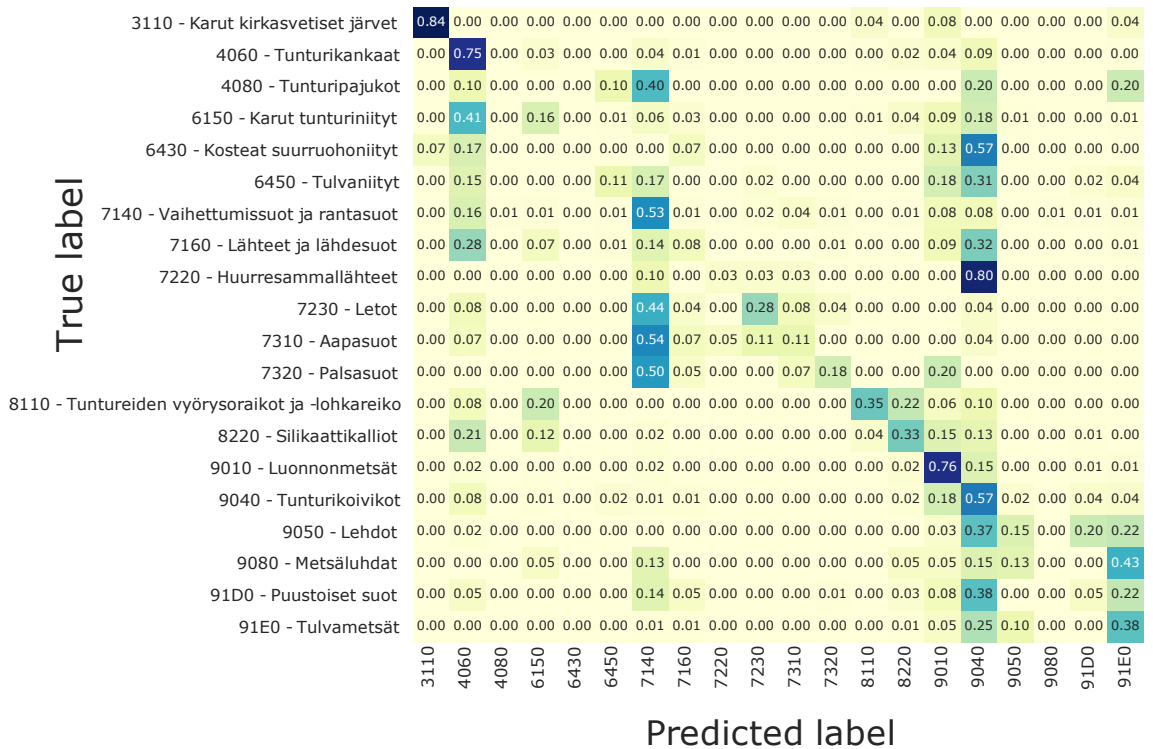
Figure 7.4. Natura2000 classes precision-recall and ROC curve comparisons between CORINE-pretrained ResNet, random forest, and ensemble models with test-time augmentation applied five times (TTA5)

The confusion matrices in Figures 7.5, 7.6 and C.1 are very similar to each other, concentrating classifications to a few classes. For Natura2000 classes, most different wetland types are classified to the class "7140 - Vaihettumissuot ja rantasuot", and smaller classes are often falsely classified to the largest class "9040 - Tunturikoivikot". Most aapa mires (7310) are classified as transition mires (7140). Interestingly, only small portions of classes are classified as palsa mires (7320), in the qualitative review of the classification maps a large portion of wetlands are classified to this class. This can be seen in Figure 7.17f. As the comparison between Figures 7.5 and 7.6 shows, the ensemble model performs better than the models separated.

The GCS classes cause even more confusion in the smaller classes. A significant portion of all classes is classified to the class mesic moss-dwarf-shrub class 242, probably because it is by far the most common class in the dataset. A real problem in this is that a large amount of wetlands are also classified in this class. The easiest classes to classify are the mossy fells (271) and the dryish lichen-moss-dwarf-shrub cover (241).



(a) Random forest



(b) ResNet

Figure 7.5. Normalized confusion matrices for Natura2000 classes using random forest and a CORINE-pretrained ResNet model with test-time augmentation applied

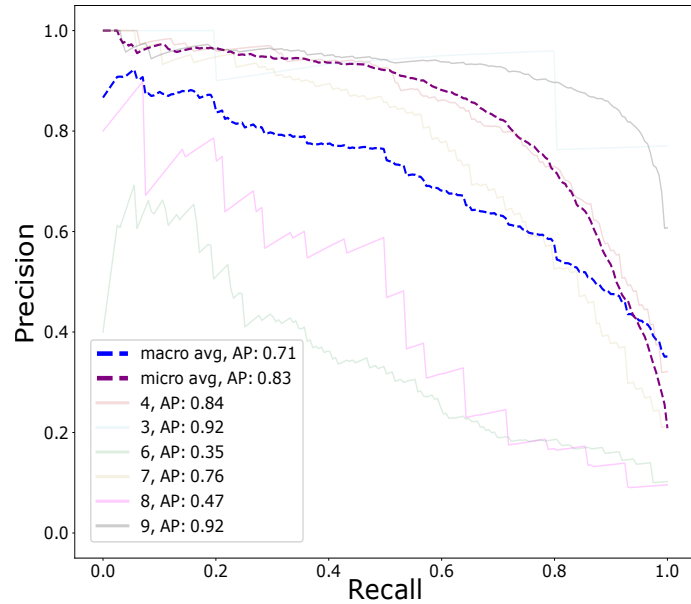
True label	3110	4060	4080	6150	6430	6450	7140	7160	7220	7230	7310	7320	8110	8220	9010	9040	9050	9080	91D0	91E0
3110 - Karut kirkasvetiset järvet	0.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.00	0.00
4060 - Tunturikankaat	0.00	0.82	0.00	0.01	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.03	0.08	0.00	0.00	0.00	0.00
4080 - Tunturipajukot	0.00	0.20	0.00	0.00	0.00	0.00	0.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.30	0.00	0.00	0.00	0.20
6150 - Karut tunturiniityt	0.00	0.41	0.00	0.21	0.00	0.00	0.06	0.03	0.00	0.00	0.00	0.00	0.01	0.03	0.08	0.15	0.01	0.00	0.00	0.01
6430 - Kosteet suurruohoniityt	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.07	0.00	0.00	0.00	0.07	0.63	0.10	0.00	0.00	0.00
6450 - Tulvaniityt	0.00	0.17	0.00	0.04	0.00	0.09	0.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	0.26	0.00	0.00	0.02	0.02
7140 - Vaihettumissuot ja rantasuot	0.00	0.18	0.00	0.02	0.00	0.01	0.57	0.00	0.00	0.01	0.02	0.02	0.00	0.01	0.08	0.07	0.00	0.01	0.01	0.01
7160 - Lähteet ja lähdesuot	0.00	0.31	0.00	0.07	0.00	0.00	0.16	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.31	0.00	0.00	0.00	0.02
7220 - Huurresammallähteet	0.00	0.03	0.00	0.00	0.00	0.00	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00
7230 - Letot	0.00	0.12	0.00	0.04	0.00	0.00	0.48	0.04	0.00	0.28	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7310 - Aapasuot	0.00	0.07	0.00	0.00	0.00	0.00	0.74	0.00	0.05	0.03	0.03	0.04	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00
7320 - Palsasuot	0.00	0.00	0.00	0.00	0.00	0.00	0.62	0.00	0.00	0.00	0.07	0.12	0.00	0.00	0.20	0.00	0.00	0.00	0.00	0.00
8110 - Tuntureiden vyörysoiraikot ja -lohkareiko	0.00	0.11	0.00	0.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.13	0.32	0.05	0.10	0.00	0.00	0.00	0.00
8220 - Silikaattikalliot	0.00	0.18	0.00	0.09	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.39	0.18	0.14	0.00	0.00	0.00	0.00
9010 - Luonnonmetsät	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.82	0.13	0.00	0.00	0.01	0.01	0.01
9040 - Tunturikoivikot	0.00	0.07	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.16	0.62	0.01	0.00	0.07	0.04	0.04
9050 - Lehdot	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.39	0.14	0.00	0.20	0.24	0.24
9080 - Metsäluhdat	0.00	0.07	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.05	0.15	0.17	0.07	0.00	0.00	0.43	0.43
91D0 - Puustoiset suot	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.05	0.00	0.00	0.00	0.01	0.00	0.03	0.11	0.45	0.00	0.00	0.04	0.20
91E0 - Tulvametsät	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.05	0.31	0.07	0.00	0.00	0.35

Figure 7.6. Normalized confusion matrix for the ensemble of models seen in Figure 7.5. The ensemble performs better than the models separately

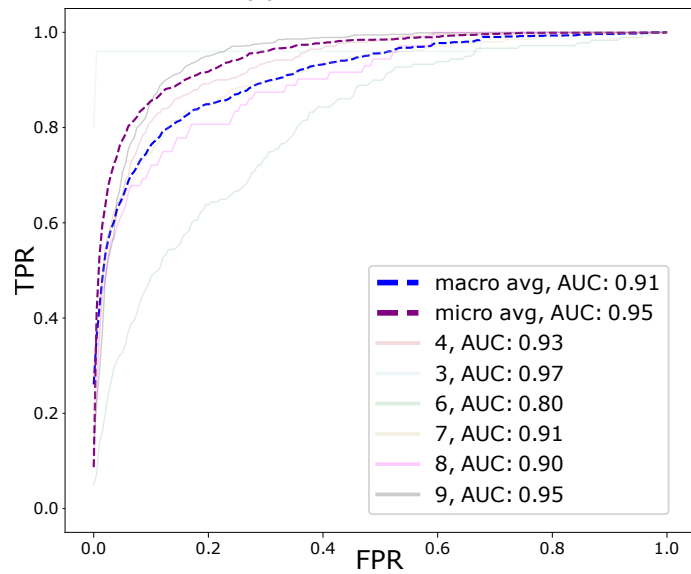
Higher hierarchies

Due to the hierarchical nature of the taxonomies, the models trained on the finest hierarchy level can be also used to evaluate the models on higher levels. Classification is still performed on the finest level, but higher level evaluations illustrate the scale of possible mistakes. Misclassifications of similar classes, for example between mire types, are not as serious than for example mistaking forests as grasslands. Similar figures to the previous ones were calculated on a higher classification hierarchy for both Natura2000 and GCS classes. To conserve space, the GCS class results for the higher hierarchy are displayed in the Appendix. The numbers for the Natura2000 classes correspond to the first values in their classification identifier: 3 - Freshwater habitats, 4 - Temperate heath and scrub, 6 - Natural and semi-natural grassland formations, 7 - Raised bogs and mires and fens, 8 - Rocky habitats and caves and 9 - Forests.

Figure 7.7 shows the precision-recall and ROC curves for each superclass. Forests and freshwaters are by far the most reliable classes to classify, while performance on the grasslands is very poor. The confusion matrices in Figure 7.9 (GCS in C.3) show that many classes are classified to the Temperate heath and scrub superclass that hosts the most populous class 4060 - Alpine and Boreal heaths. The comparison curves in Figure 7.8 indicate similar results as lower hierarchy counterparts - the ensemble model with test-time augmentation performs significantly better than other models, but the random forest is not far behind.

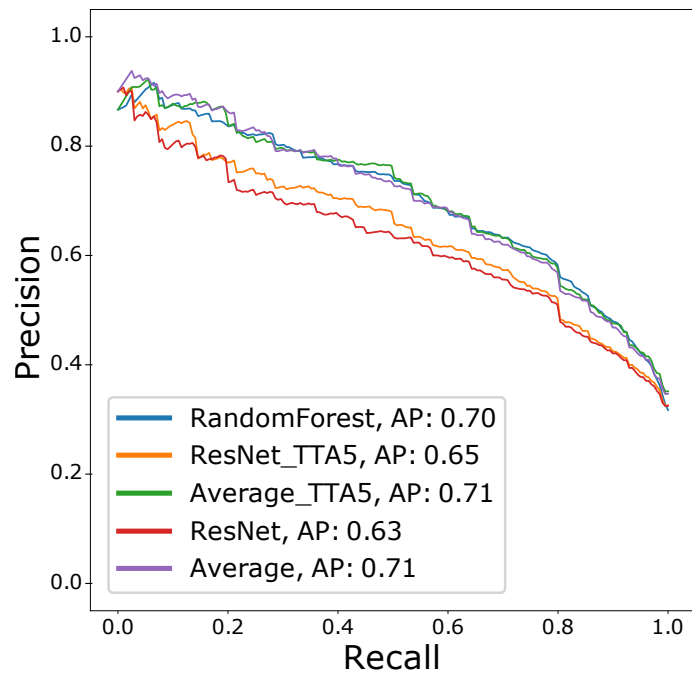


(a) Precision-recall



(b) ROC-curve

Figure 7.7. Highest hierarchy level class-wise precision-recall and ROC curves for Natura2000 classes. The numbers correspond to the Natura2000 class number's first values: 3 - Freshwater habitats, 4 - Temperate heath and scrub, 6 - Natural and semi-natural grassland formations, 7 - Raised bogs and mires and fens, 8 - Rocky habitats and caves, 9 - Forests



(a) Macro average precision-recall curves

Figure 7.8. Highest hierarchy level Natura2000 classes precision-recall curve comparisons between CORINE-pretrained ResNet, random forest and augmentation models with and without test-time augmentation applied

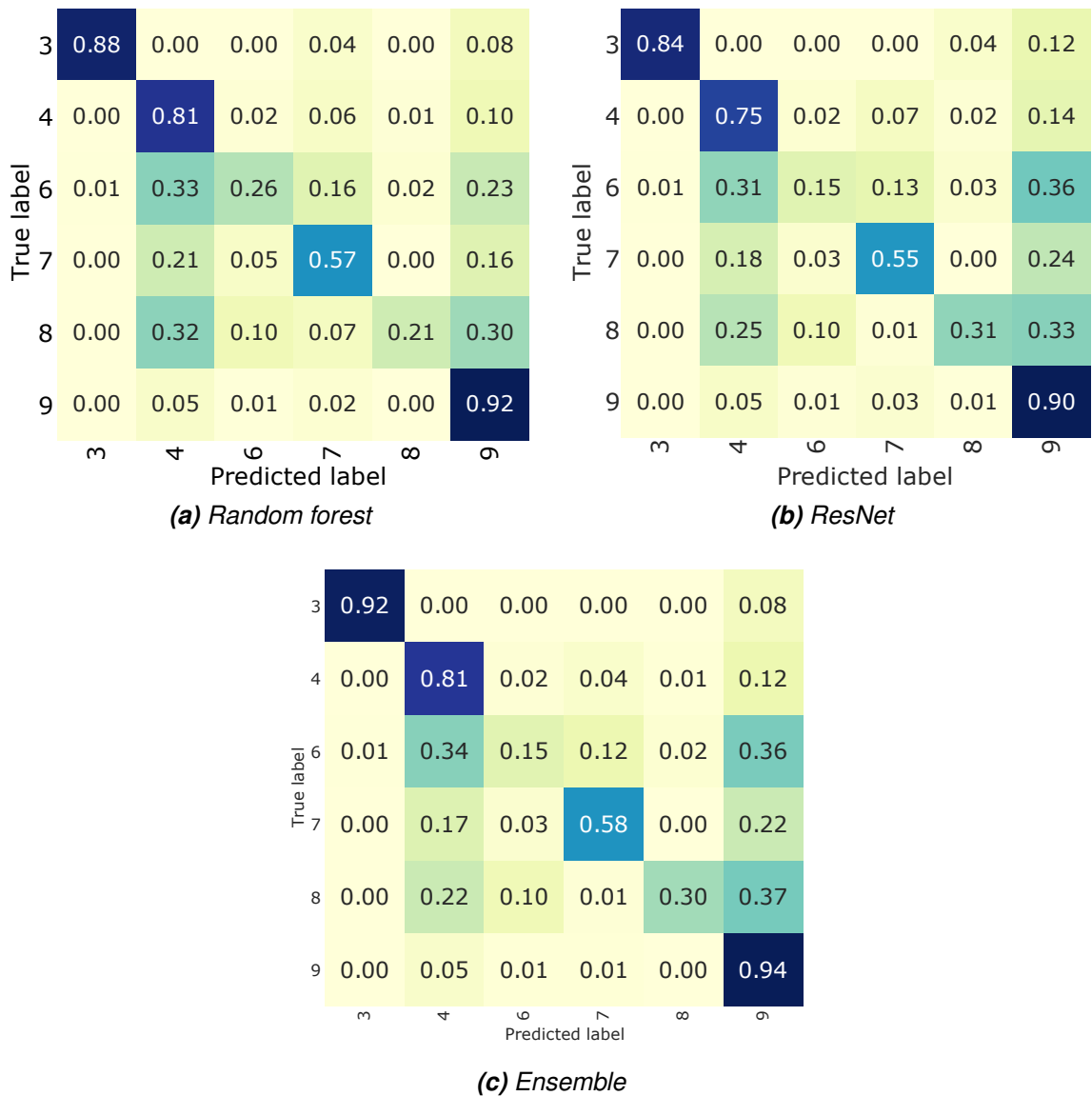


Figure 7.9. Highest Natura2000 hierarchy level normalized confusion matrices. The numbers correspond to the Natura2000 class number's first values: 3 - Freshwater habitats, 4 - Temperate heath and scrub, 6 - Natural and semi-natural grassland formations, 7 - Raised bogs and mires and fens, 8 - Rocky habitats and caves, 9 - Forests

7.1.1 Sensitivity studies

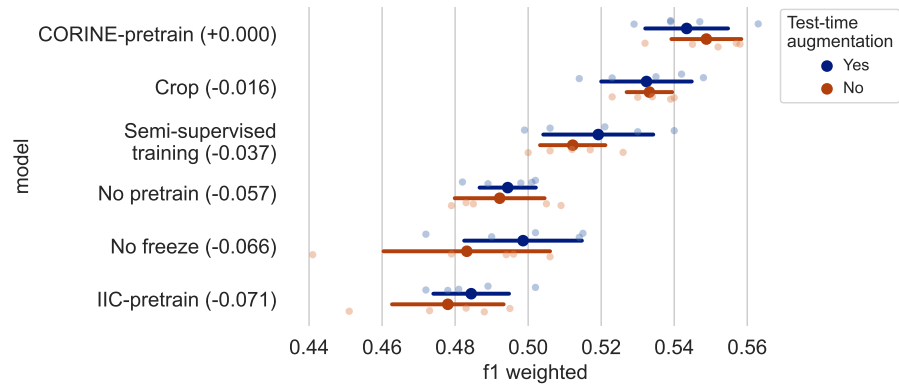
Different training methods can have a large effect on the final outcome of the classifier. Four different hypotheses were tested:

1. Pretraining using a large dataset with a coarse class taxonomy improves classification with fine-grained labels.
2. Semi-supervised learning with the larger dataset (without labels) improves performance from the pre-trained classifier
3. Training only the classification head with the fine-grained labels leads to a more generalized model
4. Augmentation by random center cropping improves performance in this specific scenario.

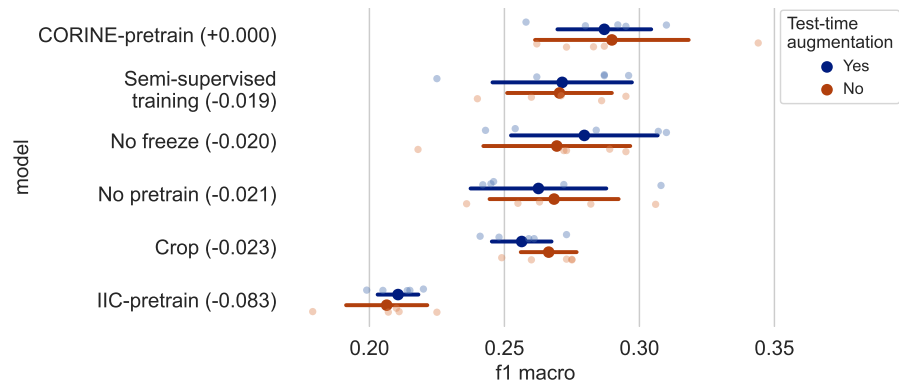
Figures 7.10 and 7.11 illustrate the effect of different model ablations and additions on the best performing model, which is the plain transfer-learned model with CORINE pretraining, without crop augmentations and with convolutional layers frozen after pretraining. Freezing refers to the practice where convolutional layer weights are not calculated during training, and only the fully connected classification head weights are learned during the transfer learning phase. The figure shows the weighted and macro F1 scores of each cross-validation fold, with the mean and single standard deviation range highlighted in bold. The effect of adding a certain attribute to the best performing model is shown under the best model performance. Test-time augmentation can be applied for each model at test time and is plotted separately for each attribute.

It can be seen that both crop augmentation and semi-supervised learning lead to a worse performing model, compared to this single model. The differences in macro-averaged F1 score are larger due to the nature of macro-averaging, where the score is biased towards smaller classes with few examples. Contrary to the original hypothesis, that semi-supervised learning and cropping would lead to the best performance, the best performing model applies neither of these. Layer freezing and pretraining however improve the model, as the model does not overfit the convolutional layers to the small dataset, and learns better representations from the CORINE-dataset.

The effect of pretraining can be also seen in Figures 7.13 and 7.12, where the effect of each attribute is plotted separately against all of the other models. Mean and standard deviation over the cross-validation folds of all models where an attribute is used, are plotted as well as the differences between having the attribute and not having it. If all models with a certain attribute are taken into account, all four methods (pretraining, convolutional layer freezing, semi-supervised learning, and cropping) improve the performance slightly. The relationships between the attributes and their effect on models are complex and would need more extensive modeling and testing, since as seen from previous figures, for a



(a) F1 weighted



(b) F1 macro

Figure 7.10. Sensitivity study for attributes affecting the best Natura2000 CORINE-pretrained model, with test-time augmentation effect plotted separately. Each scatterplot point is a result of a cross-validation fold. Large points are mean of all cross-validation folds, with the standard deviation as a bold line.

single model pipeline the semi-supervised training can also hurt performance. Only in the big picture the models with semi-supervised training perform slightly better than ones without it.

The main takeaway from the sensitivity studies is that pretraining the model with a larger dataset, even though with different domain labels, has a huge improvement on the model performance. Pretraining the model with CORINE labels has a far larger effect than other training tricks. Also, test-time-augmentation should be used during inference if the computing resources allow that.

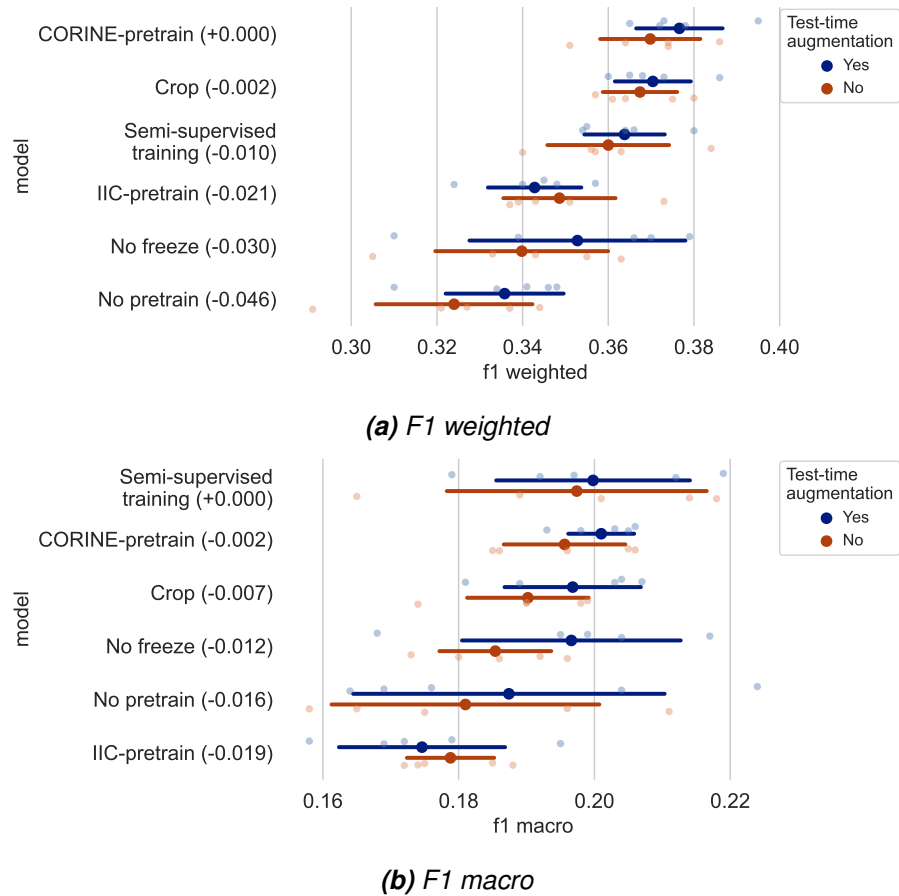
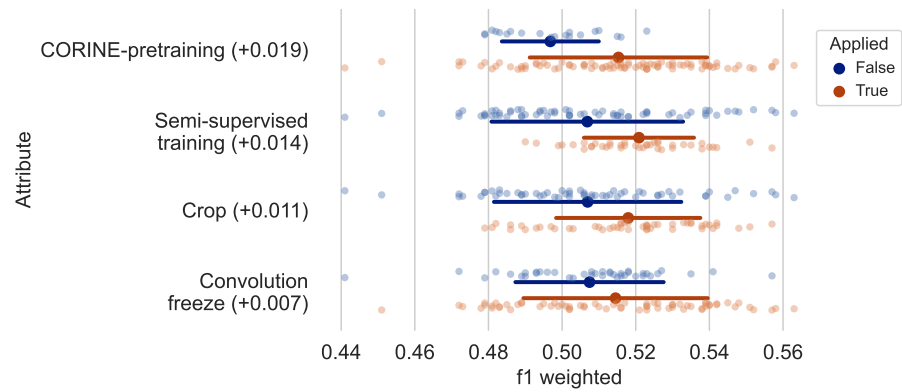
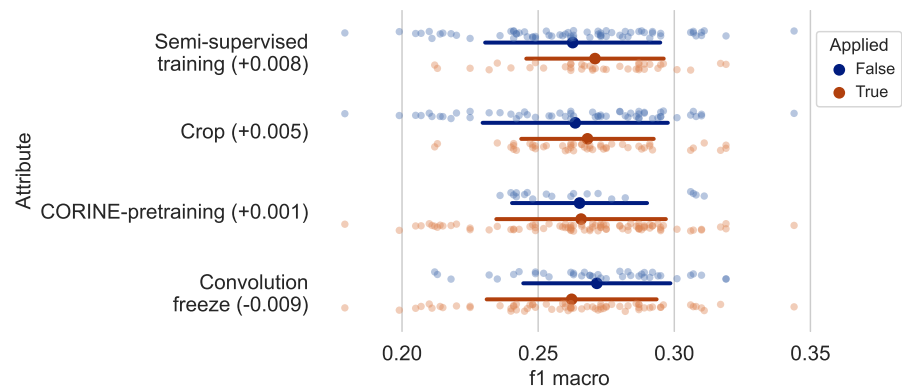


Figure 7.11. Sensitivity study for attributes affecting the best GCS CORINE-pretrained model, with test-time augmentation effect plotted separately. Each scatterplot point is a result of a cross-validation fold. Large points are mean of all cross-validation folds, with the standard deviation as a bold line. Note that in this case, with GCS classes and macro F1 score, semi-supervised training improves the classification accuracy.



(a) F1 weighted



(b) F1 macro

Figure 7.12. Natura2000 training attribute comparison for each attribute separately, with the improvement between attribute being applied or not. Each scatterplot point is a result of a cross-validation fold. Large points are mean of all cross-validation folds, with the standard deviation as bold line.

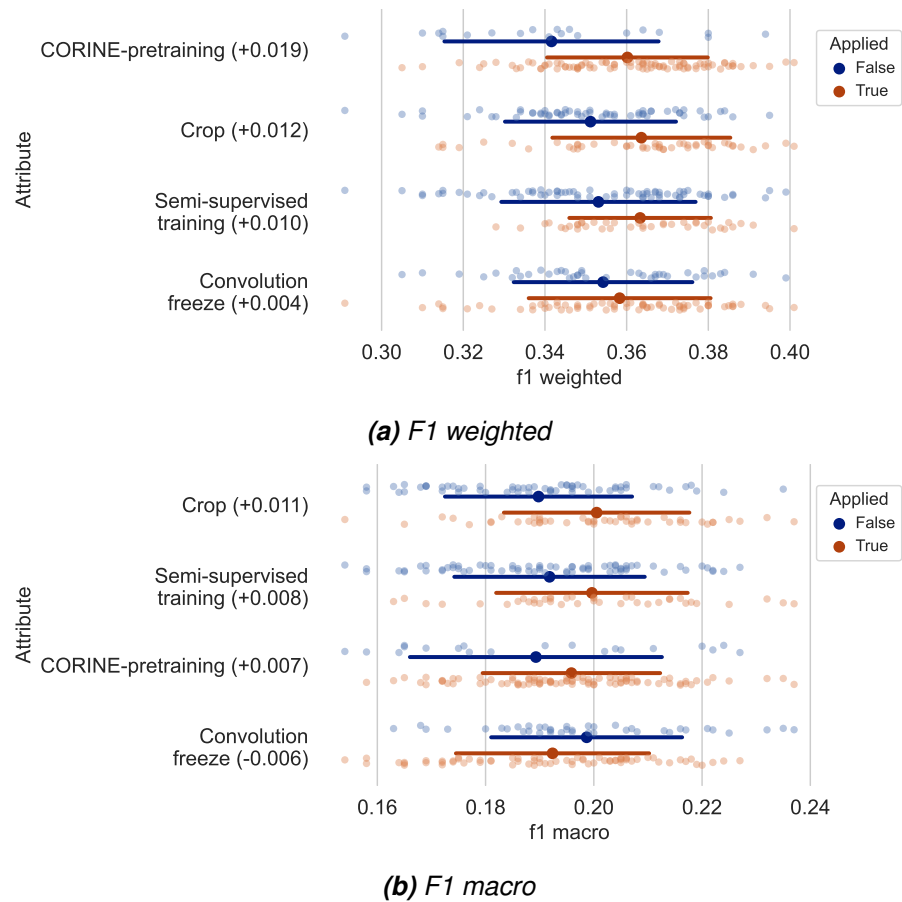


Figure 7.13. GCS training attribute comparison for each attribute separately, with the improvement between attribute being applied or not. Each scatterplot point is a result of a cross-validation fold. Large points are mean of all cross-validation folds, with the standard deviation as a bold line.

7.1.2 Classification maps

The quantitative analysis of the results can give absolute information on the dataset available, but does not give a full picture of the classifier performance. In addition to building and evaluating the machine learning models presented in this thesis, a classification map of the entire northern Lapland area was produced. The final classification maps were done using three models: the CORINE pretrained ResNet model (PT in the tables), the random forest model, and the ensemble model by averaging the outputs of these two models. Both Natura2000 and GCS classification maps were produced. Examples of the classification maps can be seen in Figure 7.14. The color legends for these maps can be found in the Appendix in Figures A.3 and A.4. Two sites are used in the visualizations in this chapter. First area is east of the Lätäseno river, with diverse characteristics of wetlands, fell habitats in the north, and forests in the south. The second sample area is the Saana fell, with high altitude changes and rare herb-rich forest and fell grassland habitats.

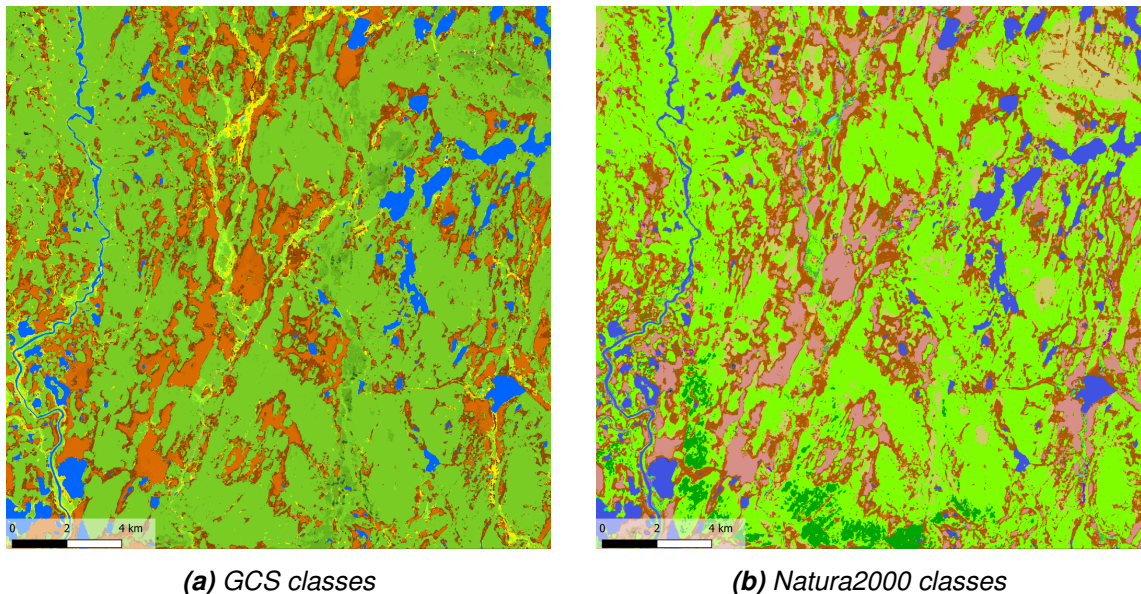


Figure 7.14. Classification maps for an area east of Lätäseno river, using the ensemble model. Color legends can be found from the Appendix.

Figure 7.15 shows the differences between the three model approaches for both taxonomies. The differences between the models are quite large, the main difference being the more uniform classification map of the ResNet model compared to the fragmentation in the random forest's map. Ensembling these models combines these features with some tradeoff.

In general, ResNet detects larger and more mosaicked areas, such as wetlands, better, while losing some accuracy in finer-grained areas with rapid land-cover changes. Some rarer classes are found only in small areas and may not be detected by the ResNet model. The windowed nature of the ResNet classifier leads to misclassifications in transitional

areas, such as in rivers and surrounding wetlands. Often, when the center pixel to be classified hits a river, it is classified into a class surrounding the river, as seen around the river area in Figure 7.15b. The ResNet model is also more eager to classify areas containing hay species as *Poaceae* and *Carex* (GCS classes 26X).

In a high-altitude habitat near the Saana fell, the differences between models are also large. Figure 7.16 shows well the differences between two taxonomies. The GCS classes focus on land cover and show how eutrophic the terrain is, while the Natura2000 classes are on a higher abstraction level, just describing habitat types as a whole. Most of the area north of the fell is classified as 4060 - Alpine and Boreal heaths, with some grassland areas appearing. The GCS classifications separate dwarf-shrub, mossy and lichen-dominated areas, which can be hard to distinguish from space. The classifications in this area also show the resolution differences between a CNN approach and a pixel-based approach. The ResNet classification areas are more uniform due to the 19x19 window, and fast changes in the environment can be lost. The ResNet, however, is good at detecting areas with complex textures, such as the rocky slopes and landslide areas northeast of the fell, seen as black in the image.

In the Natura2000 classes, a peculiar and large difference is seen between ResNet and random forest models, where most uniform wetland areas are classified as palsa mires by the ResNet model. This is especially present in the wetland close-up in Figure 7.17. Palsa mires are rare mire types with permafrost mounds and are of high interest for conservation. The ResNet model however tries to classify almost all large wetland areas, that most likely should be aapa mires, into palsa mires. This bias could be assessed by giving less weight to these classifications in the ensemble model. Overall, the ResNet model again performs well in the mosaicy and textured multi-pixel features that wetlands form, producing uniform classifications compared to the random forest model.

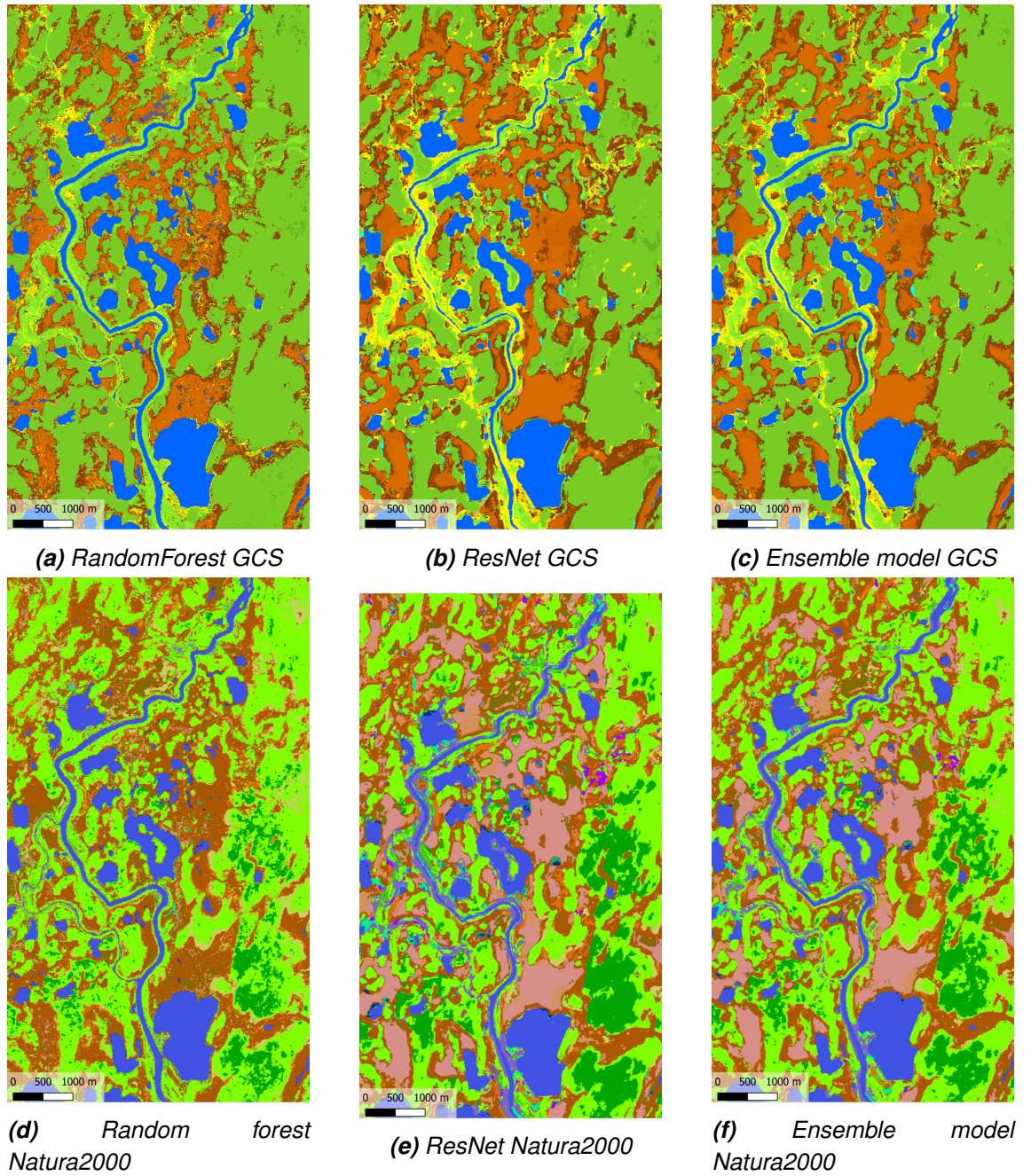
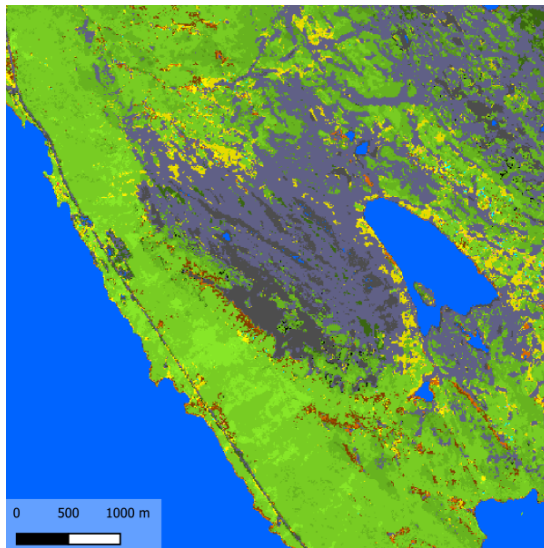
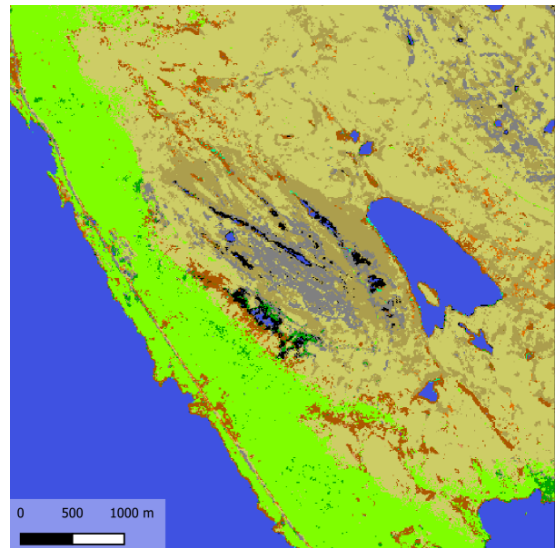


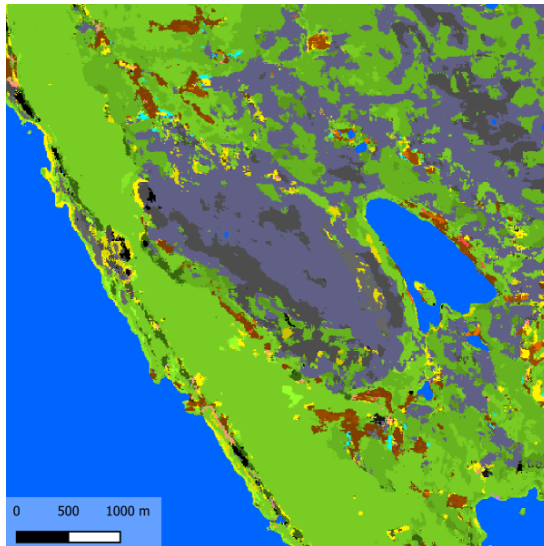
Figure 7.15. Classification maps of an area near Lätäseno river for all model types and class taxonomies.



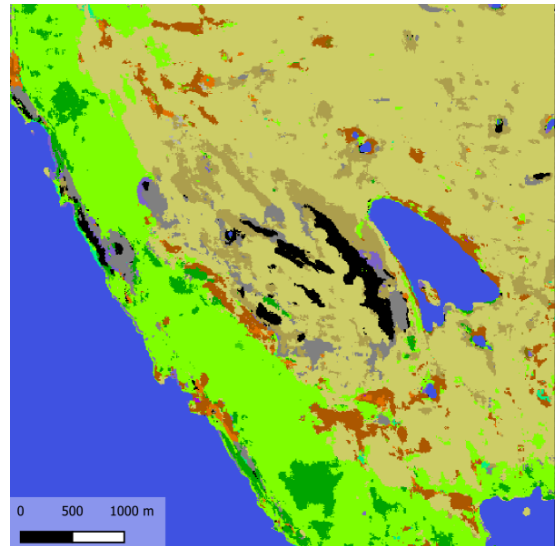
(a) Random forest GCS



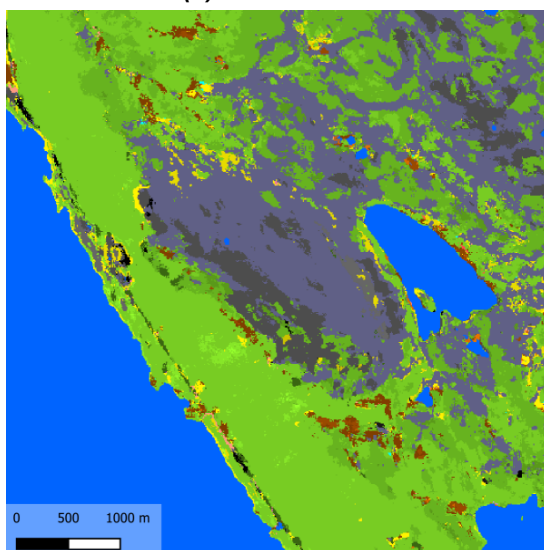
(b) Random forest Natura2000



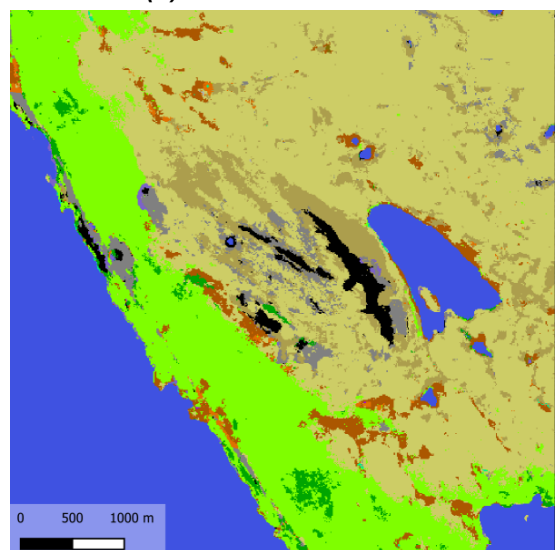
(c) ResNet GCS



(d) ResNet Natura2000

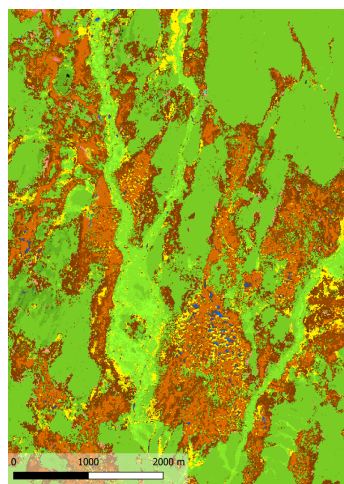


(e) Ensemble model GCS

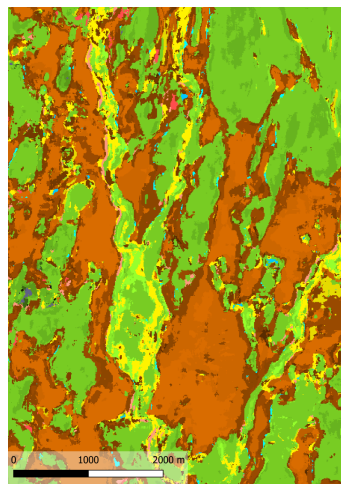


(f) Ensemble model Natura2000

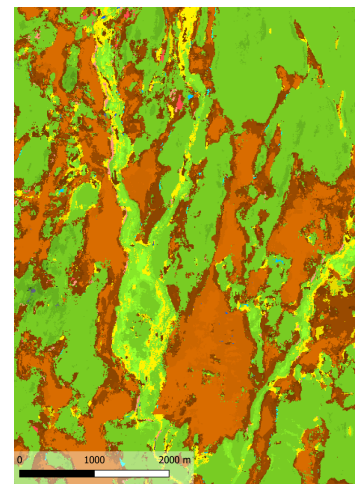
Figure 7.16. Classification maps of the Saana fell area for all model types and class taxonomies.



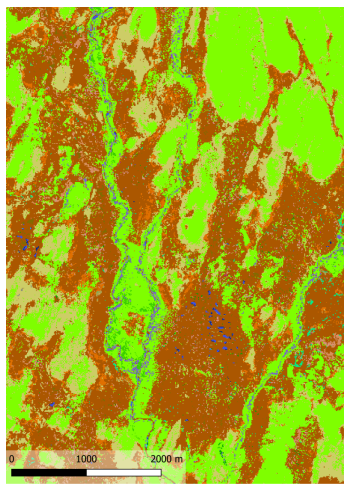
(a) *Random forest GCS*



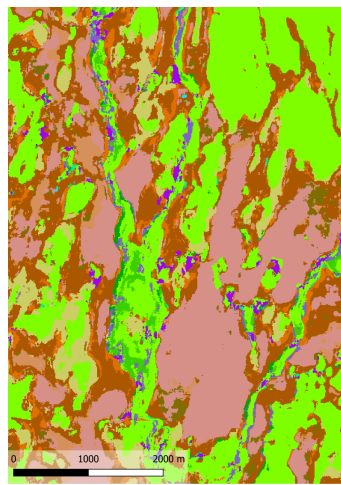
(b) *ResNet GCS*



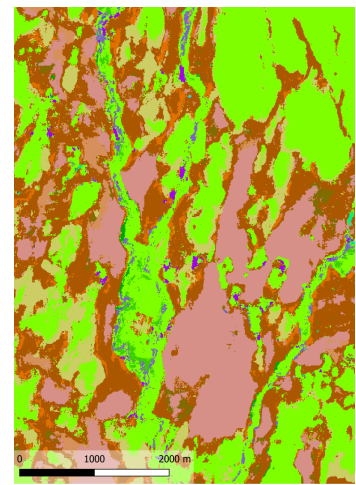
(c) *Ensemble model GCS*



(d) *Random forest
Natura2000*



(e) *Random forest
Natura2000*



(f) *Ensemble model
Natura2000*

Figure 7.17. *Close-up classification maps of an wetland area near Lätäseno river for all model types and class taxonomies.*

Confidence maps

The models can output not just classification class, but a probability distribution over all classes in the taxonomy. Using this probability distribution, it is possible to map a single class' classification confidence over the whole study area. This information can be valuable when researchers are trying to find rare nature types from new areas. Even lower confidences of a class might indicate presence in the area. Figure 7.18 shows an example of the three most common Natura2000 classes in the Lätäseno area. Alpine and Boreal heaths (4060) are shown in red, fell birch forests (9040) in blue, and Taiga forests (9010) in green.

Figure 7.19 shows the classification confidence heatmaps for the GCS class "252 Grass (herb-rich forest)" south of Saana fell. It is known that the southern slope of the fell has a high amount of herb-rich forest vegetation, while other areas nearby are more oligotrophic. Although the confidence is fairly low, the environmentally interesting herb-rich forest concentrations are visible in the class maps.

Figure 7.20 shows how the classification confidence for wetland areas is higher using the ResNet model. Only a small portion of the wetlands is classified as aapa mires, since most large areas are thought to be palsa mires by the ResNet model. Random forest is not a good classifier for the wetlands, classifying most as transition mires or grass/hay-dominated areas. The aapa mire classification probability is very low for the entire area.

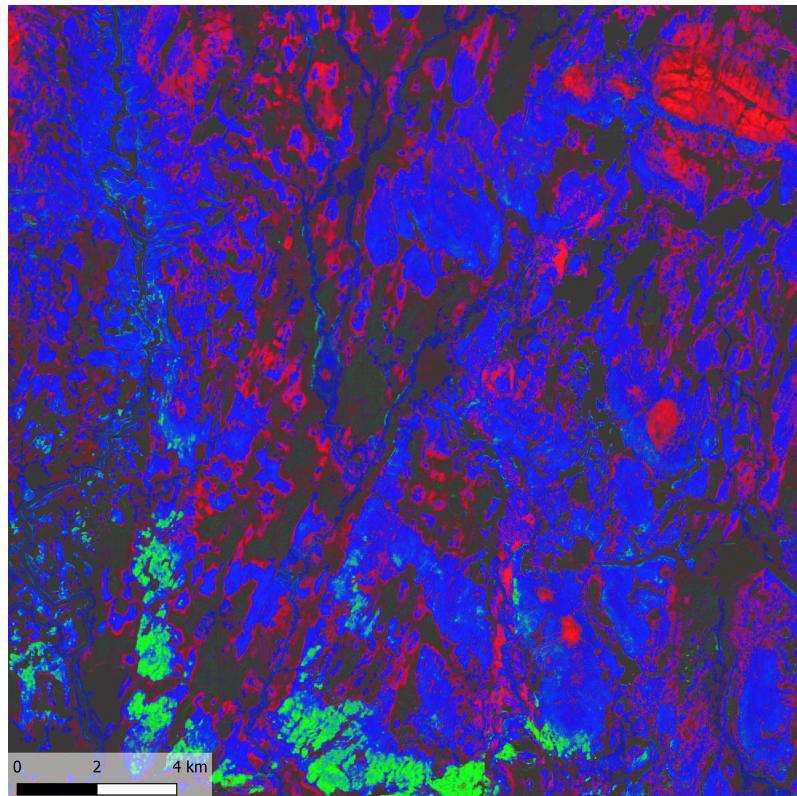


Figure 7.18. Class confidence scores for three Natura2000 classes, 4060 in red, 9010 in green, and 9040 in blue.

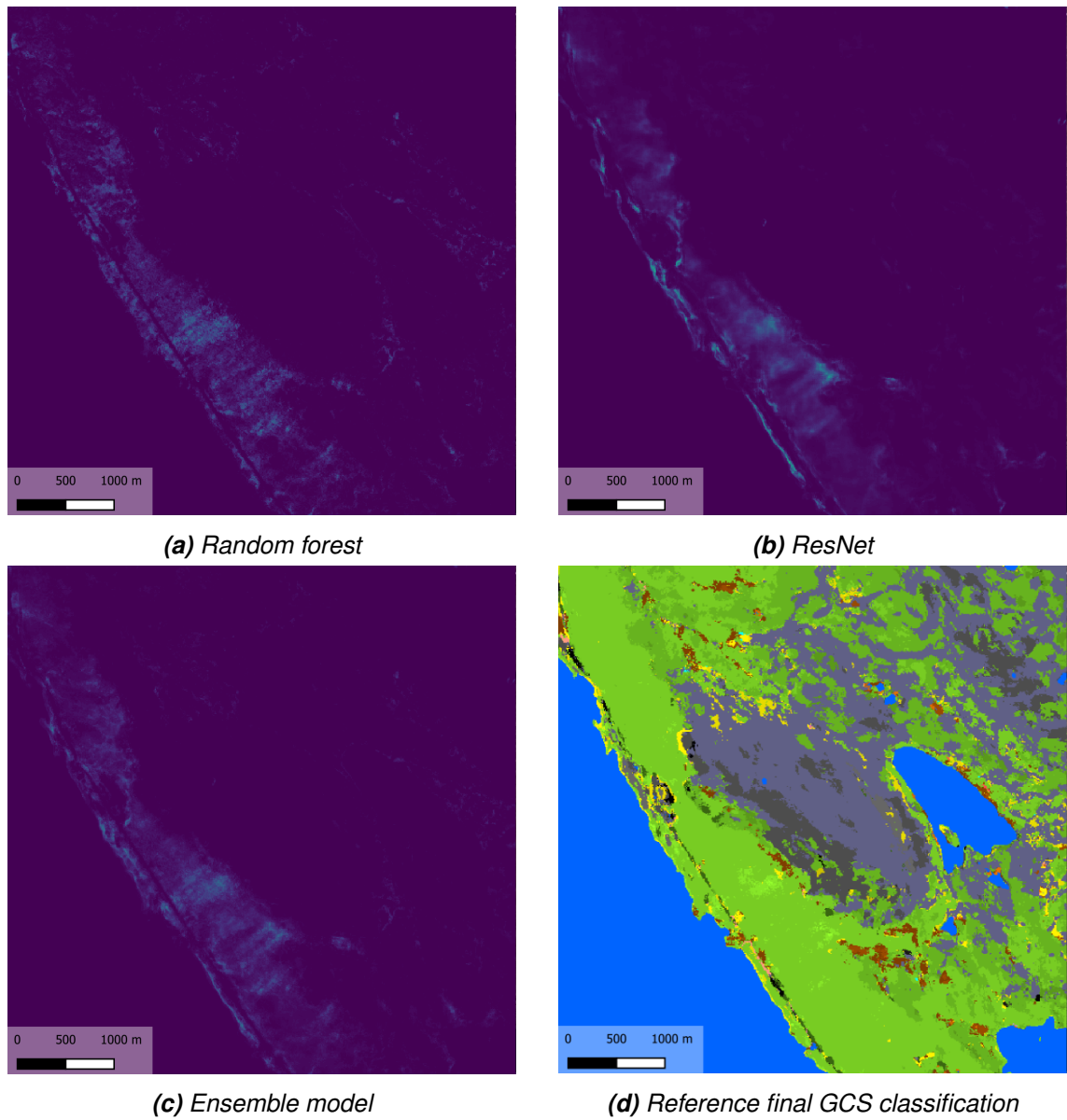
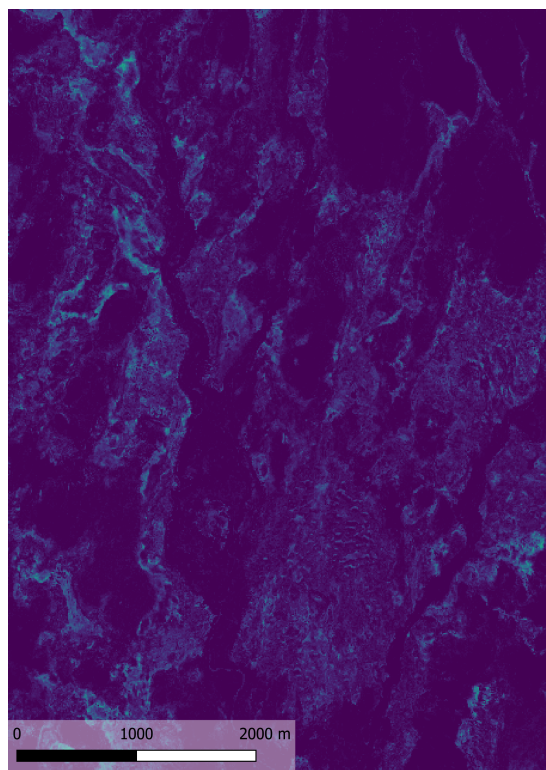
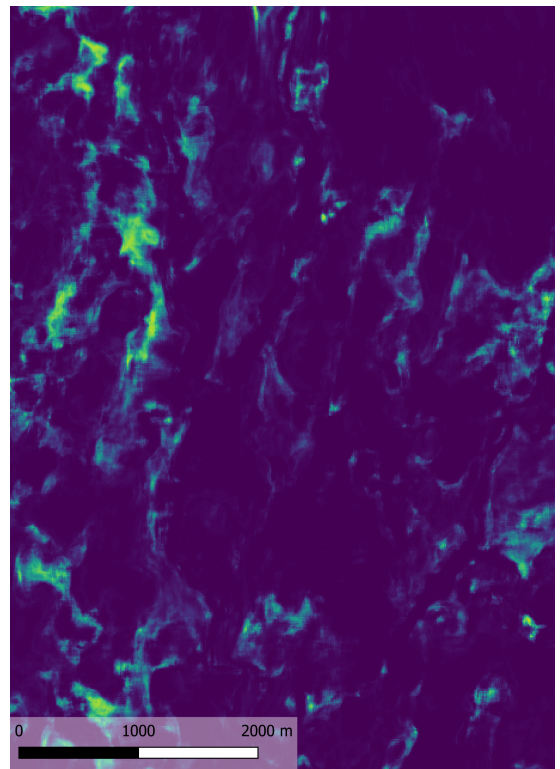


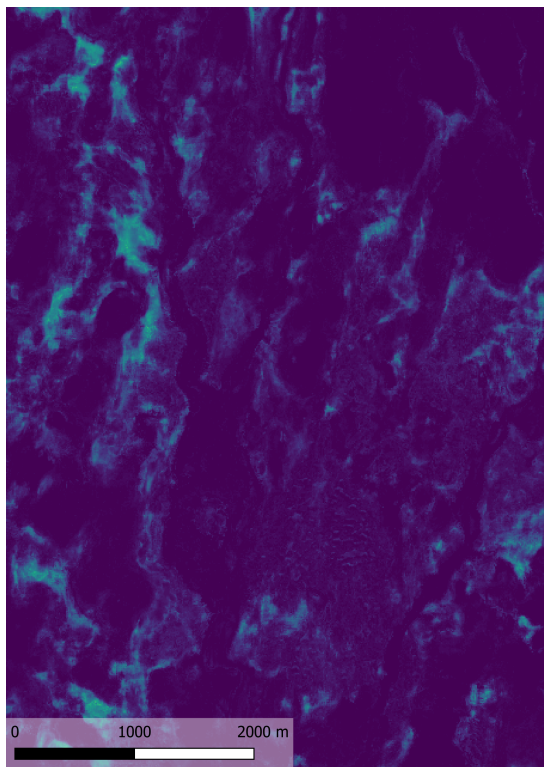
Figure 7.19. Classification heatmaps for the GCS class "252 Grass (herb-rich forest)" at Saana fell, with the ensemble classification map as reference.



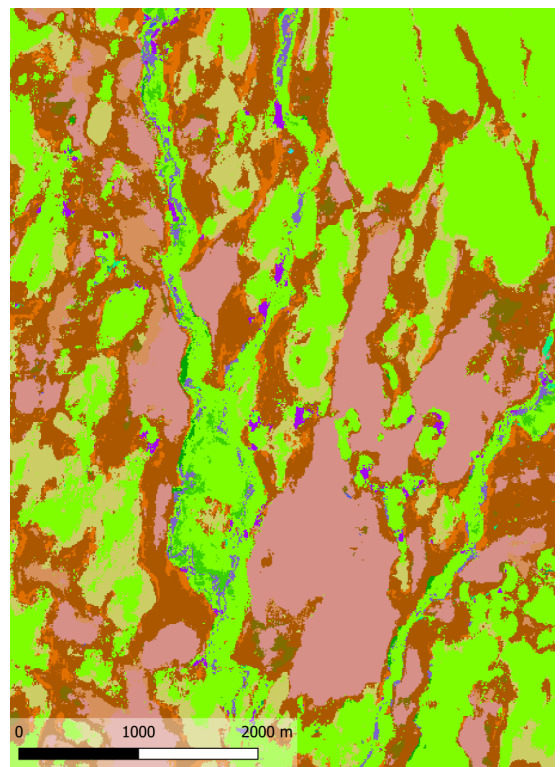
(a) ResNet



(b) Random forest



(c) Ensemble model



(d) Reference Natura2000 classification

Figure 7.20. Classification heatmaps for the Natura2000 class "7310 Aapa mires" at wetland area east of Lätäseno river, with the ensemble classification map as reference. Brighter values indicate higher confidence for the classification.

The class with the maximum confidence is chosen as the final classification for each pixel. In pixels that are harder to classify, the confidence is more distributed among classes, and the maximum class gets a lower confidence score. Mapping the maximum confidence for each pixel, like in Figure 7.21, it is possible to compare areas where the model is more confident with areas where confidence is low.

The outputs of the random forest and CNN are very different due to the nature of the models. Random forest's bagging approach produces robust and general models also with imbalanced data, making the classification confidences more distributed among classes than the ResNet models. The CNN model has very high confidences in strange areas, making some of the classifications unstable. The ensemble model balances the characteristics of both models.

In practice, the final models are usable for habitat classification with some limitations. Easy and abundant classes, such as Alpine and Boreal heaths (4060 Natura2000) or taiga forests (9010 Natura2000) are easy to detect and the mapping results are reliable. With smaller classes detection accuracy falls and the amount of false positives and negatives makes classification of these classes unreliable. A suitable practical approach could be to remove unreliable classes from the classification, or map them to a single class and focus only on distinguishing the easy classes. Unfortunately the small classes are usually the most important ones that scientists are most interested in detecting from the wilderness areas. Finding more training data for smaller classes with a classifier that is tuned for a high recall rate (leading to a lot of false positives), could be used to find suitable field-collection sites.

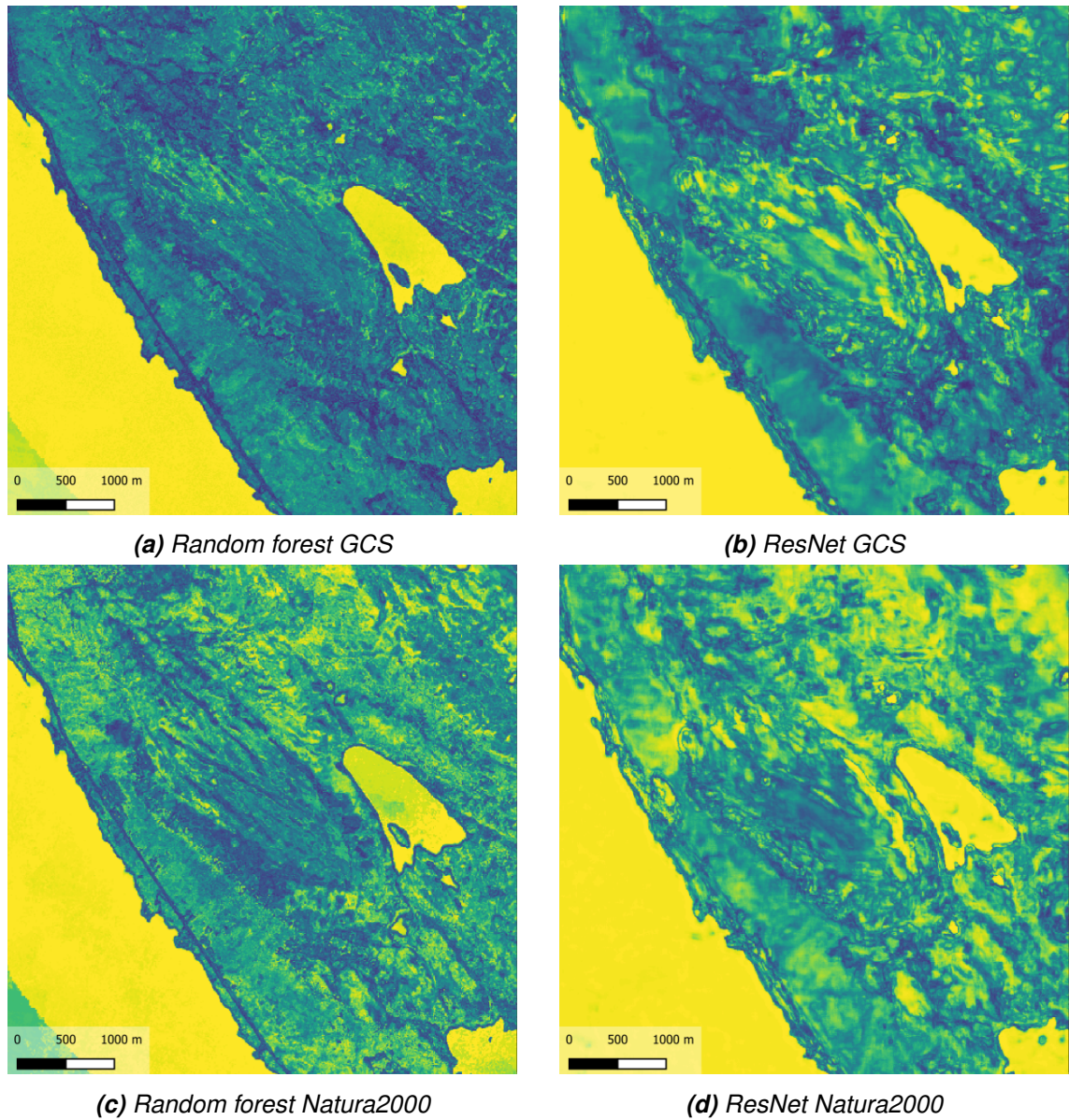


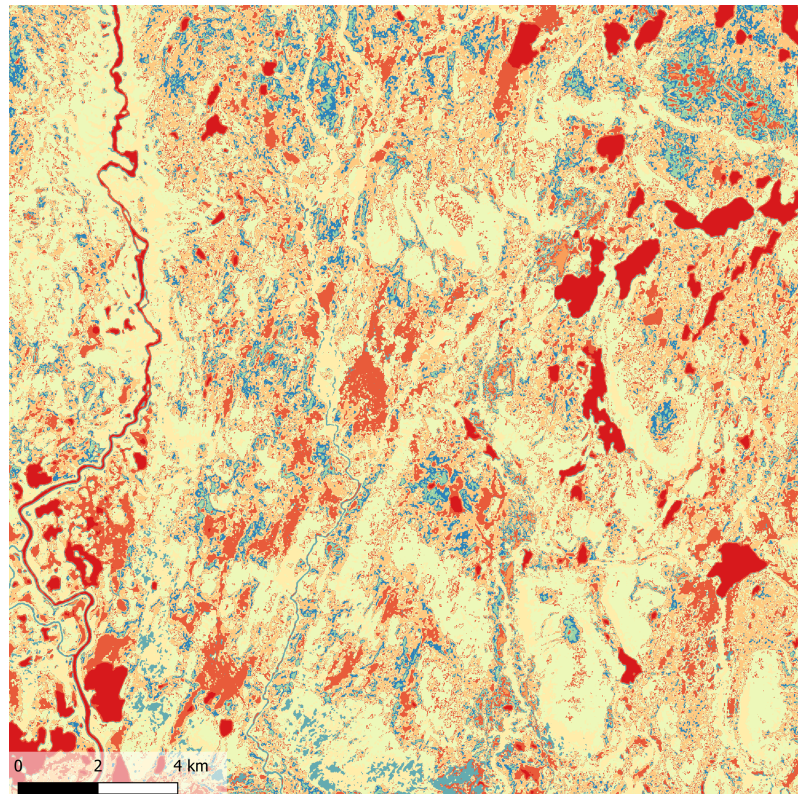
Figure 7.21. Maximum class confidence maps of the Saana fell area. Brighter values indicate higher confidence for the most confident class, while darker values indicate that even the maximum confidence is very low.

7.2 Unsupervised segmentation

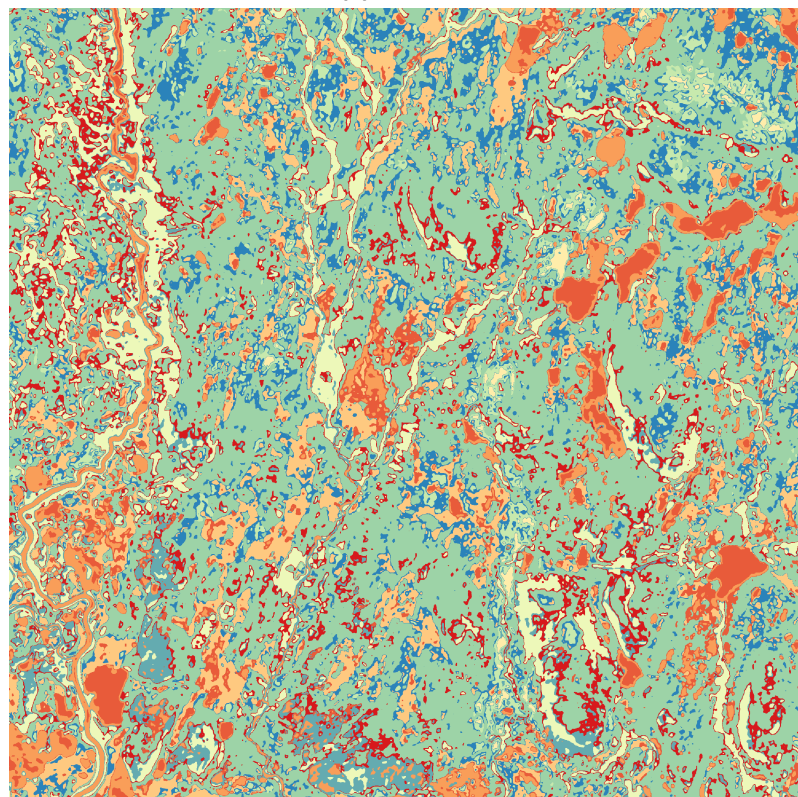
Fully unsupervised segmentation was proposed as an approach to land cover mapping in Chapter 4. This method does not need any ground truth data and cannot be biased to the chosen taxonomy. A pixel is clustered based on a $N \times N$ window around it. Because of this, the chosen window size has a large effect on the results. Figure 7.22 shows the difference between a 10 group clustering between 3×3 and 9×9 window sizes in a large scale, and Figure 7.23 in detail.

A characteristic of the approach of clustering the entire window area is that different "transitional areas" are clustered as their own clusters. For example the areas where the entire 3×3 window is inside a river is a separate cluster, but the riverbank, where half of the window is river and half is something else, is a separate cluster. This is especially visible in the larger 9×9 window.

The IIC clustering, however, is able to learn surprisingly good clusters that correspond to semantic meanings humans give to areas. Wetlands, with their mosaic characteristics are usually in their own cluster, and oligotrophic areas in the fells, seen in Figure 7.24 are separated as well. The outputs of the 30 cluster 3×3 IIC model in Figure 7.24e and the GCS supervised random forest model in Figure 7.16a are surprisingly similar, although the IIC has zero guidance other than the source data.

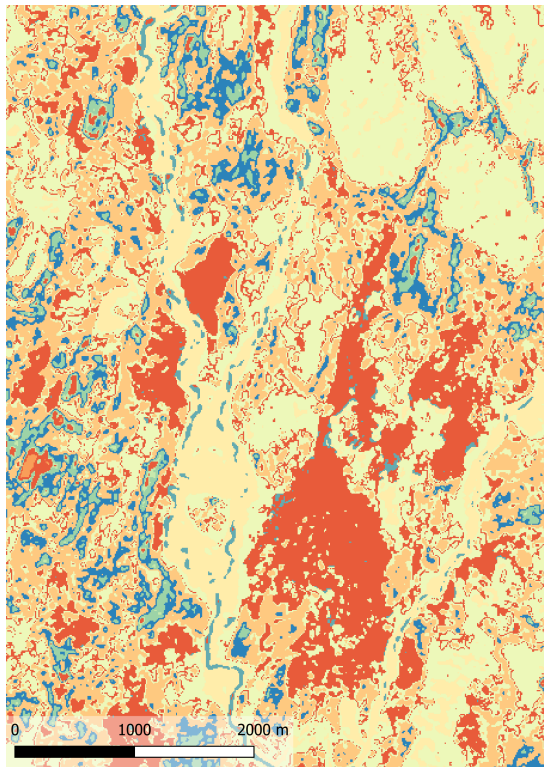


(a) 3x3 10c

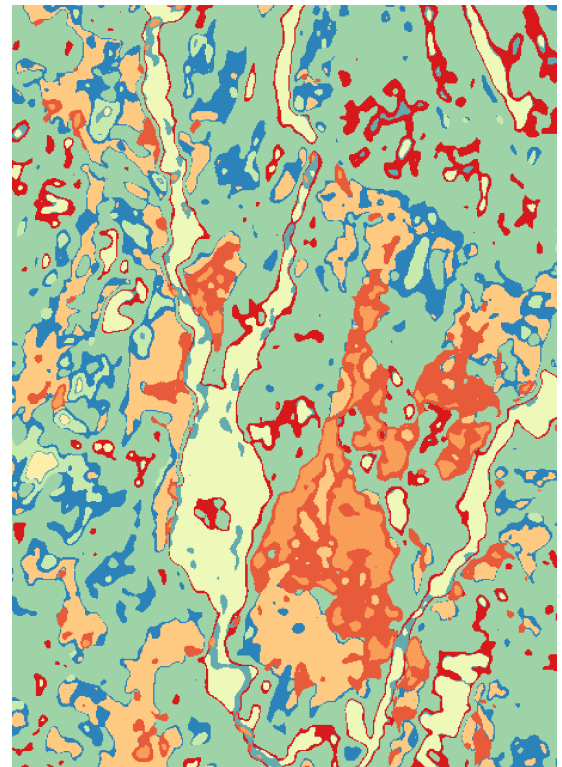


(b) 9x9 10c

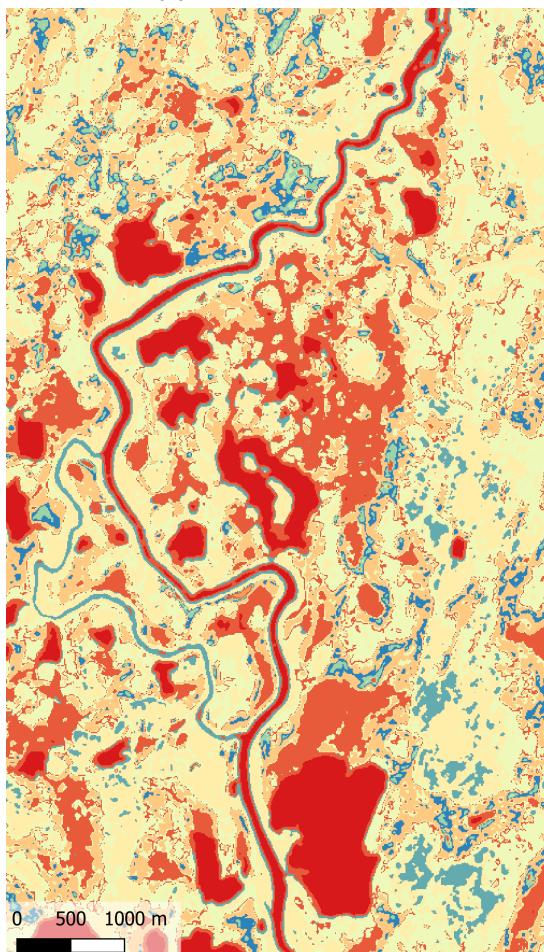
Figure 7.22. Clustering for 3x3 and 9x9 windows east of Lätäseno river, with 10 clusters.



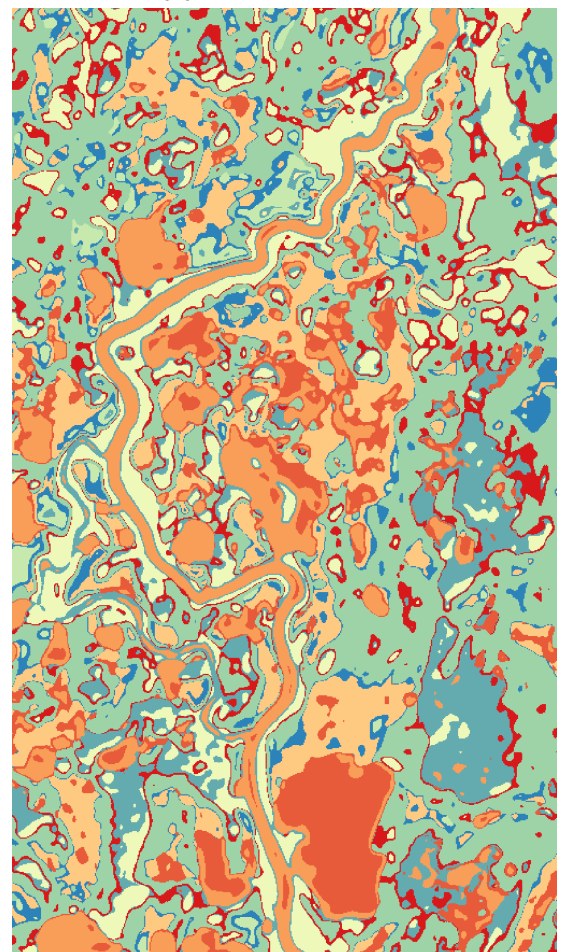
(a) 3x3 10c wetlands



(b) 9x9 10c wetlands

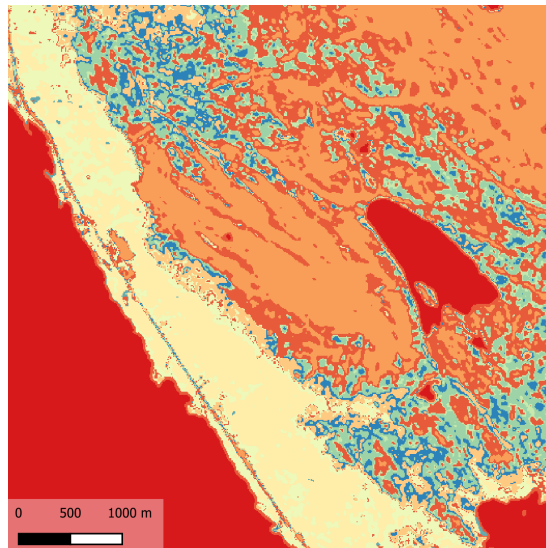


(c) 3x3 10c river

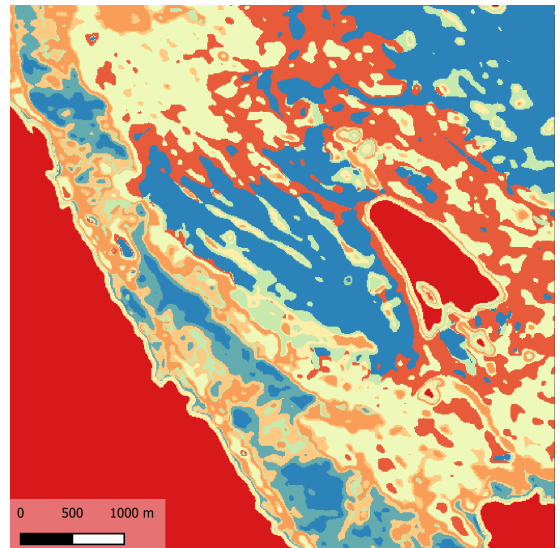


(d) 9x9 10c river

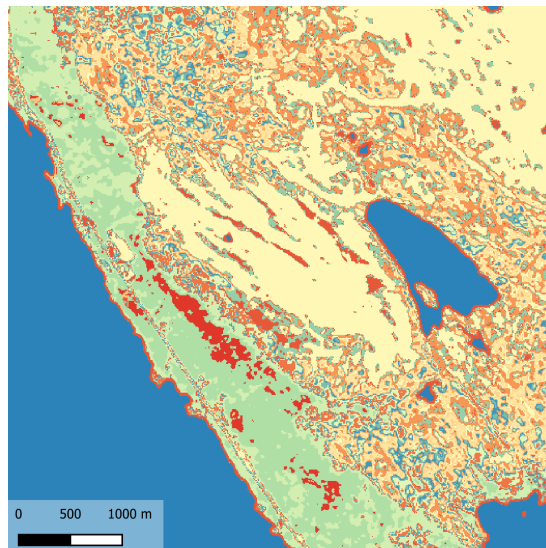
Figure 7.23. Clustering close-ups for 3x3 and 9x9 windows for 10 clusters.



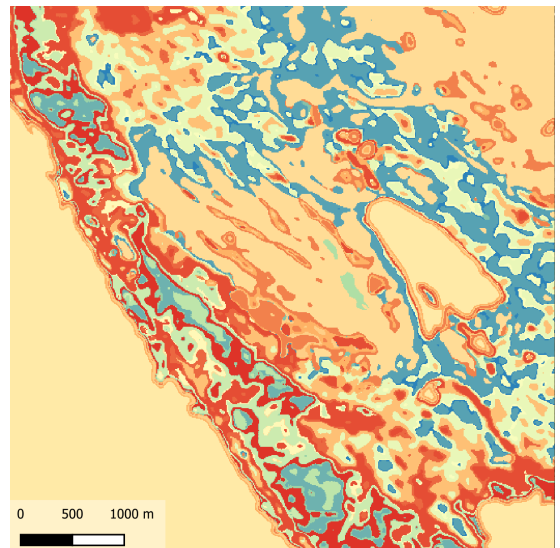
(a) 3x3 10c



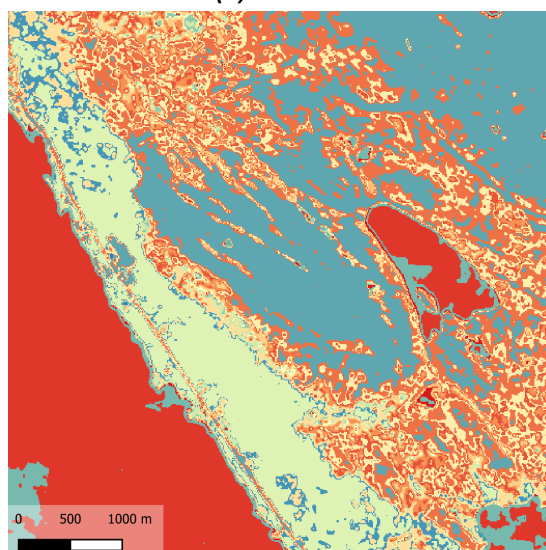
(b) 9x9 10c



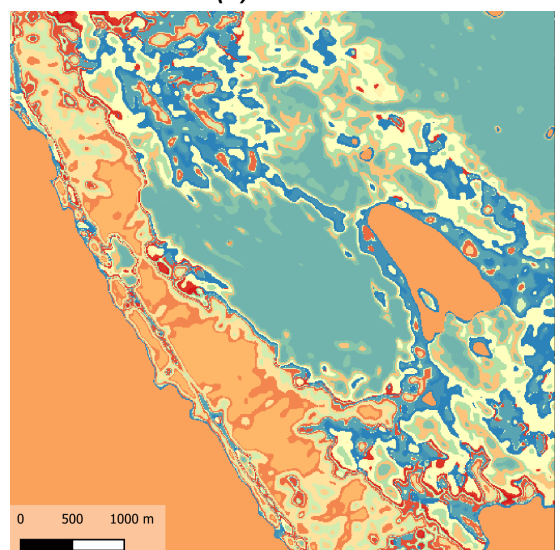
(c) 3x3 70c



(d) 9x9 70c



(e) 3x3 30c



(f) 9x9 30c

Figure 7.24. Clustering comparison for 3 different cluster amounts, 10, 30 and 70, with 3x3 and 9x9 window sizes at the Saana fell.

8. CONCLUSION

This thesis presented a habitat classification approach using an ensemble of CNNs and random forests, with remote sensed raster data as input. Methods for tackling the problem of sparse annotation were presented, and a combination of pixel-based and object-based classification was proposed. As a result, a classification map of the entire northern Lapland area was produced for two different classification taxonomies.

The results were assessed by comparing the plain CNN and random forest models to the ensemble model. Several different combinations of training approaches were tested, including transfer learning, semi-supervised learning, and unsupervised pretraining. The ensemble model outperformed single models in almost all cases. The best performing model turned out to be a model with convolutional layers trained with a large CORINE dataset, and the classification head fine-tuned with the final dataset. Class-wise comparison shows that the performance difference across classes is significant, with some classes performing very poorly and some classes being easy to classify.

Two different augmentation approaches were proposed: a random center cropping augmentation, that uses the knowledge that the unit of classification is the center pixel of a patch and using test-time augmentation during inference. Overall, across all tested models, random cropping produced slight improvement to the results. Test-time augmentation turned out to be very useful, with significant improvement on all classes.

Unsupervised and semi-supervised approaches were also tested. Semi-supervised training with a teacher-student distillation approach performed on par with a plain transfer learning model, but by analysing the overall improvement across different training approaches, it can be seen that semi-supervised learning usually improves the performance of a model. Unsupervised pretraining and fully unsupervised segmentation were tested both quantitatively and qualitatively. Unsupervised pretraining and fine-tuning the classifier with a small dataset turned out to perform poorly. However, fully unsupervised training and segmentation provided good results, by learning and detecting complex areas such as wetlands.

The methods presented in this thesis can be applied in further field work. Confidence maps produced by the models can be used as a heuristic in choosing future sampling sites. Remote sensing technologies and automatic habitat mapping provide tools for ex-

perts. Automatic models provide a good starting point in classifying and mapping the vast wilderness of northern Lapland, a task that would be extremely time-consuming by field-surveying.

As for machine learning research, this thesis shows the difficulty of a fine-grained classification problem, where the semantic meaning is inferred from a different input (field-work) than the final prediction (remote sensed imagery). Semantic differences in taxonomies produce very different results, as classes in some taxonomies can be easier to distinguish from each other. Current machine learning research focuses often on fairly easy taxonomies, such as the ImageNet 1000 classes [86], where the classes are everyday objects and common animals. A positive trend is that fine-grained datasets, such as the iNaturalist dataset [87], are gaining attention. The methods presented in this thesis that are applied to sparse, point-like data in remote sensed imagery datasets can also be applied to other similar problems, where training data is scarce but data itself is abundant. Model explainability remains a challenge, and further research would be needed to produce models where the reasoning behind a classification can be reliably explained.

REFERENCES

- [1] Kitamori, K., Manders, T., Dellink, R. and Tabeau, A. *OECD environmental outlook to 2050: the consequences of inaction*. Tech. rep. OECD, 2012.
- [2] Barnosky, A. D., Matzke, N., Tomiya, S., Wogan, G. O., Swartz, B., Quental, T. B., Marshall, C., McGuire, J. L., Lindsey, E. L., Maguire, K. C. et al. Has the Earth's sixth mass extinction already arrived?: *Nature* 471.7336 (2011), pp. 51–57.
- [3] Dirzo, R. and Raven, P. H. Global state of biodiversity and loss. *Annual review of Environment and Resources* 28.1 (2003), pp. 137–167.
- [4] Hoekstra, J. M., Boucher, T. M., Ricketts, T. H. and Roberts, C. Confronting a biome crisis: global disparities of habitat loss and protection. *Ecology letters* 8.1 (2005), pp. 23–29.
- [5] Cardinale, B. J., Duffy, J. E., Gonzalez, A., Hooper, D. U., Perrings, C., Venail, P., Narwani, A., Mace, G. M., Tilman, D., Wardle, D. A. et al. Biodiversity loss and its impact on humanity. *Nature* 486.7401 (2012), pp. 59–67.
- [6] Foley, J. A., DeFries, R., Asner, G. P., Barford, C., Bonan, G., Carpenter, S. R., Chapin, F. S., Coe, M. T., Daily, G. C., Gibbs, H. K. et al. Global consequences of land use. *science* 309.5734 (2005), pp. 570–574.
- [7] Dasgupta, P. *The Economics of Biodiversity: the Dasgupta Review*. HM Treasury, 2021.
- [8] Lengyel, S., Déri, E., Varga, Z., Horváth, R., Tóthmérés, B., Henry, P.-Y., Kobler, A., Kutnar, L., Babij, V., Seliškar, A. et al. Habitat monitoring in Europe: a description of current practices. *Biodiversity and Conservation* 17.14 (2008), pp. 3327–3339.
- [9] Mumby, P., Green, E., Edwards, A. and Clark, C. The cost-effectiveness of remote sensing for tropical coastal resources assessment and management. *Journal of Environmental Management* 55.3 (1999), pp. 157–166.
- [10] Rhodes, C. J., Henrys, P., Siriwardena, G. M., Whittingham, M. J. and Norton, L. R. The relative value of field survey and remote sensing for biodiversity assessment. *Methods in Ecology and Evolution* 6.7 (2015), pp. 772–781.
- [11] Petrou, Z. and Petrou, M. A review of remote sensing methods for biodiversity assessment and bioindicator extraction. *2011 2nd International Conference on Space Technology*. Sept. 2011, pp. 1–5.
- [12] Wang, J., Jiang, L., Wang, Y. and Qi, Q. An Improved Hybrid Segmentation Method for Remote Sensing Images. en. *ISPRS International Journal of Geo-Information* 8.12 (Dec. 2019). Number: 12 Publisher: Multidisciplinary Digital Publishing Institute, p. 543.

- [13] Kentsch, S., Lopez Caceres, M. L., Serrano, D., Roure, F. and Diez, Y. Computer Vision and Deep Learning Techniques for the Analysis of Drone-Acquired Forest Images, a Transfer Learning Study. *Remote Sensing* 12.8 (Jan. 2020). Number: 8 Publisher: Multidisciplinary Digital Publishing Institute, p. 1287.
- [14] Mahdavi, S., Salehi, B., Granger, J., Amani, M., Brisco, B. and Huang, W. Remote sensing for wetland classification: a comprehensive review. *GIScience & Remote Sensing* 55.5 (Sept. 2018), pp. 623–658.
- [15] Yuan, Q., Shen, H., Li, T., Li, Z., Li, S., Jiang, Y., Xu, H., Tan, W., Yang, Q., Wang, J. et al. Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sensing of Environment* 241 (2020), p. 111716.
- [16] Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P. et al. Sentinel-2: ESA's optical high-resolution mission for GMES operational services. *Remote sensing of Environment* 120 (2012), pp. 25–36.
- [17] Kontula, T. and Raunio, A. Threatened Habitat Types in Finland 2018. Red List of Habitats – Results and Basis for Assessment. *The Finnish Environment* 2 (2019), p. 254.
- [18] Xie, Q., Luong, M.-T., Hovy, E. and Le, Q. V. Self-training with noisy student improves imagenet classification. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 10687–10698.
- [19] Ji, X., Henriques, J. F. and Vedaldi, A. Invariant information clustering for unsupervised image classification and segmentation. *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 9865–9874.
- [20] Weih, R. C. and Riggan, N. D. Object-based classification vs. pixel-based classification: Comparative importance of multi-resolution imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38.4 (2010), p. C7.
- [21] Blaschke, T. Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing* 65.1 (2010), pp. 2–16.
- [22] Häme, T., Siro, L. and Kilpi, J. A Hierarchical Clustering Method for Land Cover Change Detection and Identification. *Remote Sensing* (May 2020).
- [23] Desclée, B., Bogaert, P. and Defourny, P. Forest change detection by statistical object-based method. *Remote sensing of environment* 102.1-2 (2006), pp. 1–11.
- [24] Pascual, C., García-Abril, A., García-Montero, L. G., Martín-Fernández, S. and Cohen, W. Object-based semi-automatic approach for forest structure characterization using lidar data in heterogeneous *Pinus sylvestris* stands. *Forest Ecology and Management* 255.11 (2008), pp. 3677–3685.
- [25] Duveiller, G., Defourny, P., Desclée, B. and Mayaux, P. Deforestation in Central Africa: Estimates at regional, national and landscape levels by advanced process-

- ing of systematically-distributed Landsat extracts. *Remote sensing of environment* 112.5 (2008), pp. 1969–1981.
- [26] Mäyrä, J., Keski-Saari, S., Kivinen, S., Tanhuanpää, T., Hurskainen, P., Kullberg, P., Poikolainen, L., Viinikka, A., Tuominen, S., Kumpula, T. and Vihervaara, P. Tree species classification from airborne hyperspectral and LiDAR data using 3D convolutional neural networks. en. *Remote Sensing of Environment* 256 (Apr. 2021), p. 112322.
- [27] Wang, S., Chen, W., Xie, S. M., Azzari, G. and Lobell, D. B. Weakly Supervised Deep Learning for Segmentation of Remote Sensing Imagery. en. *Remote Sensing* 12.2 (Jan. 2020). Number: 2 Publisher: Multidisciplinary Digital Publishing Institute, p. 207.
- [28] Murphy, K. P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [29] Hyde, D. and Raffman, D. Sorites Paradox. *The Stanford Encyclopedia of Philosophy*. Ed. by E. N. Zalta. Summer 2018. Metaphysics Research Lab, Stanford University, 2018.
- [30] Williamson, T. *Vagueness*. Routledge, 2002.
- [31] Hastie, T., Tibshirani, R. and Friedman, J. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.
- [32] Goodfellow, I., Bengio, Y. and Courville, A. *Deep Learning*. <http://www.deeplearningbook.org>. MIT press, 2016.
- [33] He, K., Zhang, X., Ren, S. and Sun, J. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [34] Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal* 27.3 (1948), pp. 379–423.
- [35] James, G., Witten, D., Hastie, T. and Tibshirani, R. *An introduction to statistical learning*. Vol. 112. Springer, 2013.
- [36] Schweizer, B. and Sklar, A. *Probabilistic metric spaces*. Courier Corporation, 2011.
- [37] Manolakis, D. G., Lockwood, R. B. and Cooley, T. W. *Hyperspectral imaging remote sensing: physics, sensors, and algorithms*. Cambridge University Press, 2016.
- [38] Koch, B. Status and future of laser scanning, synthetic aperture radar and hyperspectral remote sensing data for forest biomass assessment. *ISPRS Journal of Photogrammetry and Remote sensing* 65.6 (2010), pp. 581–590.
- [39] Ignatenko, V., Laurila, P., Radius, A., Lamentowski, L., Antropov, O. and Muff, D. IC-EYE Microsatellite SAR Constellation Status Update: Evaluation of first commercial imaging modes. *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2020, pp. 3581–3584.
- [40] Lim, K., Treitz, P., Wulder, M., St-Onge, B. and Flood, M. LiDAR remote sensing of forest structure. *Progress in physical geography* 27.1 (2003), pp. 88–106.

- [41] European Space Agency. *Sentinel Online*. <https://sentinels.copernicus.eu/web/sentinel/home>.
- [42] European Space Agency. *Multispectral Instrument (MSI) Overview*. <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-2-msi/msi-instrument>.
- [43] Gislason, P. O., Benediktsson, J. A. and Sveinsson, J. R. Random Forests for land cover classification. en. *Pattern Recognition Letters*. Pattern Recognition in Remote Sensing (PRRS 2004) 27.4 (Mar. 2006), pp. 294–300.
- [44] Liu, Y., Zhang, B., Wang, L.-m. and Wang, N. A self-trained semisupervised SVM approach to the remote sensing land cover classification. en. *Computers & Geosciences* 59 (Sept. 2013), pp. 98–107.
- [45] Sherrah, J. Fully Convolutional Networks for Dense Semantic Labelling of High-Resolution Aerial Imagery. en. *arXiv:1606.02585 [cs]* (June 2016). arXiv: 1606.02585.
- [46] Mulla, D. J. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosystems engineering* 114.4 (2013), pp. 358–371.
- [47] Wei, Y. and Ji, S. Scribble-Based Weakly Supervised Deep Learning for Road Surface Extraction From Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing* (2021).
- [48] Hua, Y., Marcos, D., Mou, L., Zhu, X. X. and Tuia, D. Semantic segmentation of remote sensing images with sparse annotations. *IEEE Geoscience and Remote Sensing Letters* (2021).
- [49] Laban, N., Abdellatif, B., Ebeid, H. M., Shedeed, H. A. and Tolba, M. F. Sparse Pixel Training of Convolutional Neural Networks for Land Cover Classification. *IEEE Access* 9 (2021), pp. 52067–52078.
- [50] Nalepa, J., Myller, M., Imai, Y., Honda, K.-I., Takeda, T. and Antoniak, M. Unsupervised segmentation of hyperspectral images using 3-D convolutional autoencoders. *IEEE Geoscience and Remote Sensing Letters* 17.11 (2020), pp. 1948–1952.
- [51] Mou, L. and Zhu, X. X. Learning to pay attention on spectral domain: A spectral attention module-based convolutional network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* 58.1 (2019), pp. 110–122.
- [52] McDermid, G. J., Franklin, S. E. and LeDrew, E. F. Remote sensing for large-area habitat mapping. *Progress in Physical Geography* 29.4 (2005), pp. 449–474.
- [53] Kopel, D., Michalska-Hejduk, D., Berezowski, T., Borowski, M., Rosadzifski, S., Chormafski, J. et al. Application of multisensoral remote sensing data in the mapping of alkaline fens Natura 2000 habitat. *Ecological Indicators* 70 (2016), pp. 196–208.
- [54] Dong, L., Du, H., Mao, F., Han, N., Li, X., Zhou, G., Zhu, D., Zheng, J., Zhang, M., Xing, L. and Liu, T. Very High Resolution Remote Sensing Imagery Classification Using a Fusion of Random Forest and Deep Learning Technique—Subtropical Area

- for Example. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13 (2020). Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, pp. 113–128.
- [55] Magnússon, R. Í., Limpens, J., Kleijn, D., Huissteden, K. van, Maximov, T. C., Lobbry, S. and Heijmans, M. M. Shrub decline and expansion of wetland vegetation revealed by very high resolution land cover change detection in the Siberian lowland tundra. *Science of the Total Environment* 782 (2021), p. 146877.
- [56] Günes, M. A machine learning assessment of multi-resolution remote sensing data for Natura 2000 dune habitat classification. (2020).
- [57] Ronneberger, O., Fischer, P. and Brox, T. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.
- [58] Phillips, S. J., Anderson, R. P. and Schapire, R. E. Maximum entropy modeling of species geographic distributions. *Ecological modelling* 190.3-4 (2006), pp. 231–259.
- [59] Stenzel, S., Feilhauer, H., Mack, B., Metz, A. and Schmidlein, S. Remote sensing of scattered Natura 2000 habitats using a one-class classifier. en. *International Journal of Applied Earth Observation and Geoinformation* 33 (Dec. 2014), pp. 211–217.
- [60] Laradji, I. H., Rostamzadeh, N., Pinheiro, P. O., Vázquez, D. and Schmidt, M. Instance segmentation with point supervision. *arXiv preprint arXiv:1906.06392* (2019).
- [61] Bearman, A., Russakovsky, O., Ferrari, V. and Fei-Fei, L. What’s the point: Semantic segmentation with point supervision. *European conference on computer vision*. Springer. 2016, pp. 549–565.
- [62] Bock, M., Xofis, P., Mitchley, J., Rossner, G. and Wissen, M. Object-oriented methods for habitat mapping at multiple scales—Case studies from Northern Germany and Wye Downs, UK. *Journal for Nature Conservation* 13.2-3 (2005), pp. 75–89.
- [63] Feilhauer, H., Dahlke, C., Doktor, D., Lausch, A., Schmidlein, S., Schulz, G. and Stenzel, S. Mapping the local variability of Natura 2000 habitats with remote sensing. *Applied vegetation science* 17.4 (2014), pp. 765–779.
- [64] Borre, J. V., Spanhove, T. and Haest, B. Towards a mature age of remote sensing for Natura 2000 habitat conservation: Poor method transferability as a prime obstacle. *The roles of remote sensing in nature conservation*. Springer, 2017, pp. 11–37.
- [65] Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Dohersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R. and Valko, M. Bootstrap your own latent: A new approach to self-supervised Learning. *arXiv:2006.07733 [cs, stat]* (Sept. 2020).
- [66] Chen, T., Kornblith, S., Norouzi, M. and Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. *arXiv:2002.05709 [cs, stat]* (June 2020).

- [67] Chen, T., Kornblith, S., Swersky, K., Norouzi, M. and Hinton, G. Big Self-Supervised Models are Strong Semi-Supervised Learners. *arXiv:2006.10029 [cs, stat]* (Oct. 2020).
- [68] Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C. and Krishnan, D. Supervised Contrastive Learning. *arXiv:2004.11362 [cs, stat]* (Apr. 2020).
- [69] Hénaff, O. J., Srinivas, A., De Fauw, J., Razavi, A., Doersch, C., Eslami, S. M. A. and Oord, A. v. d. Data-Efficient Image Recognition with Contrastive Predictive Coding. *arXiv:1905.09272 [cs]* (July 2020).
- [70] Chen, X., Fan, H., Girshick, R. and He, K. Improved Baselines with Momentum Contrastive Learning. *arXiv:2003.04297 [cs]* (Mar. 2020).
- [71] Weiss, K., Khoshgoftaar, T. M. and Wang, D. A survey of transfer learning. *Journal of Big data* 3.1 (2016), pp. 1–40.
- [72] Khalilia, M., Chakraborty, S. and Popescu, M. Predicting disease risks from highly imbalanced data using random forest. *BMC medical informatics and decision making* 11.1 (2011), pp. 1–13.
- [73] Zhou, L. and Wang, H. Loan default prediction on large imbalanced data using random forests. *TELKOMNIKA Indonesian Journal of Electrical Engineering* 10.6 (2012), pp. 1519–1525.
- [74] *Interpretation manual of European Union habitats*. European Commission DG Environment. 2013.
- [75] Tuominen, S., Eeronheimo, H. and Toivonen, H. *Yleispiirteinen biotooppiluokitus*. 57. 2001, p. 60.
- [76] *CORINE Land Cover*. <https://land.copernicus.eu/pan-european/corine-land-cover>.
- [77] *CORINE Land Cover 2018 dataset Finland*. "<https://ckan.ymparisto.fi/dataset/corine-maanpeite-2018>". Finnish Environment Institute SYKE, 2018.
- [78] Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [79] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J. and Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems* 32. Curran Associates, Inc., 2019, pp. 8024–8035.
- [80] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

- [81] QGIS Development Team. *QGIS Geographic Information System*. Open Source Geospatial Foundation. 2009. URL: <http://qgis.org>.
- [82] Rocklin, M. Dask: Parallel computation with blocked algorithms and task scheduling. *Proceedings of the 14th python in science conference*. Vol. 130. Citeseer. 2015, p. 136.
- [83] Biewald, L. *Experiment Tracking with Weights and Biases*. Software available from wandb.com. 2020. URL: <https://www.wandb.com/>.
- [84] Impiö, M. On imbalanced classification of benthic macroinvertebrates: Metrics and loss-functions. B.S. thesis. 2020.
- [85] Forman, G. and Scholz, M. Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *Acm Sigkdd Explorations Newsletter* 12.1 (2010), pp. 49–57.
- [86] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. and Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.
- [87] Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P. and Belongie, S. The inaturalist species classification and detection dataset. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8769–8778.

APPENDIX A: SOURCE DATA

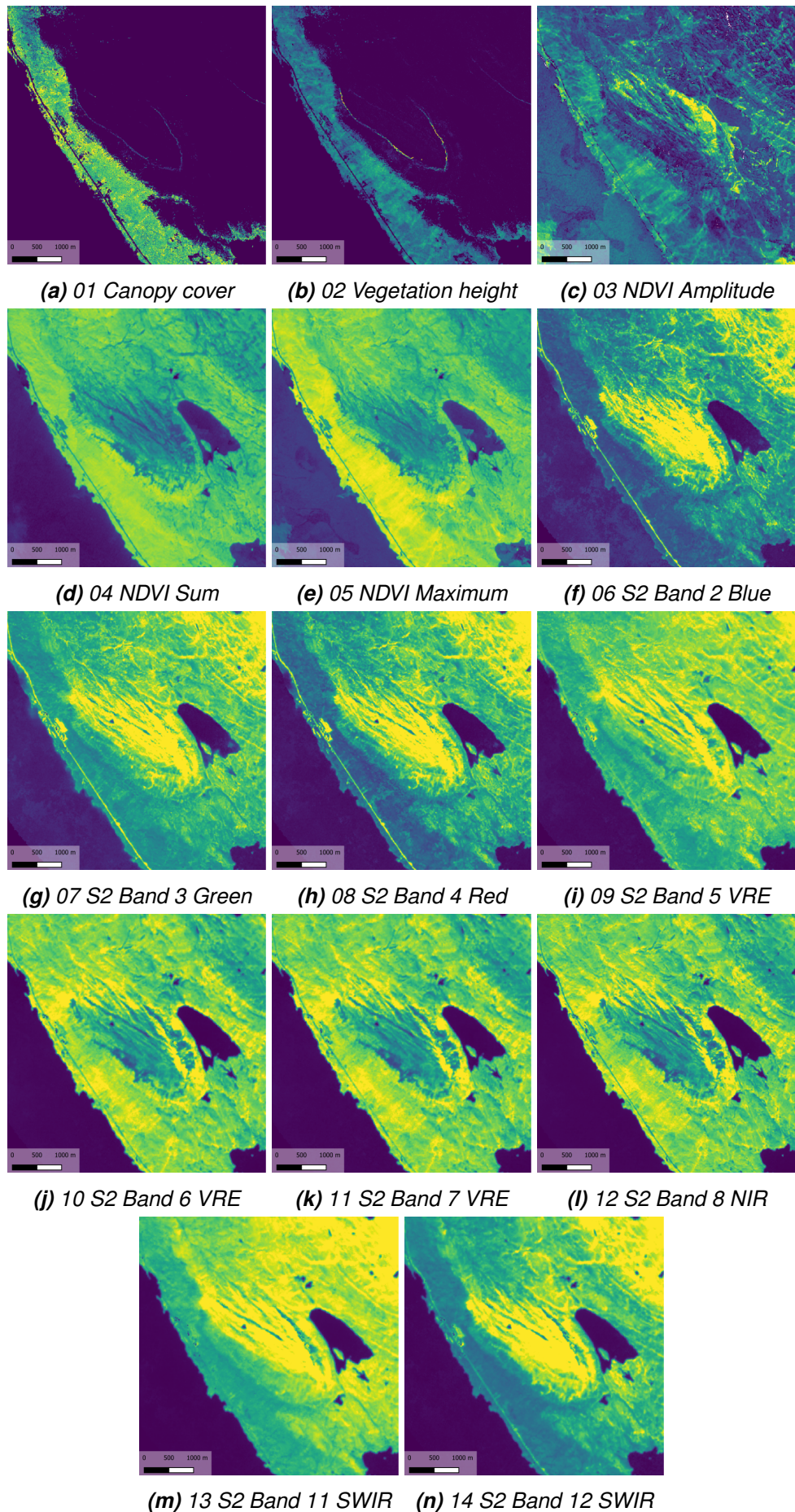


Figure A.1. Source raster channels separated for Saana fell area

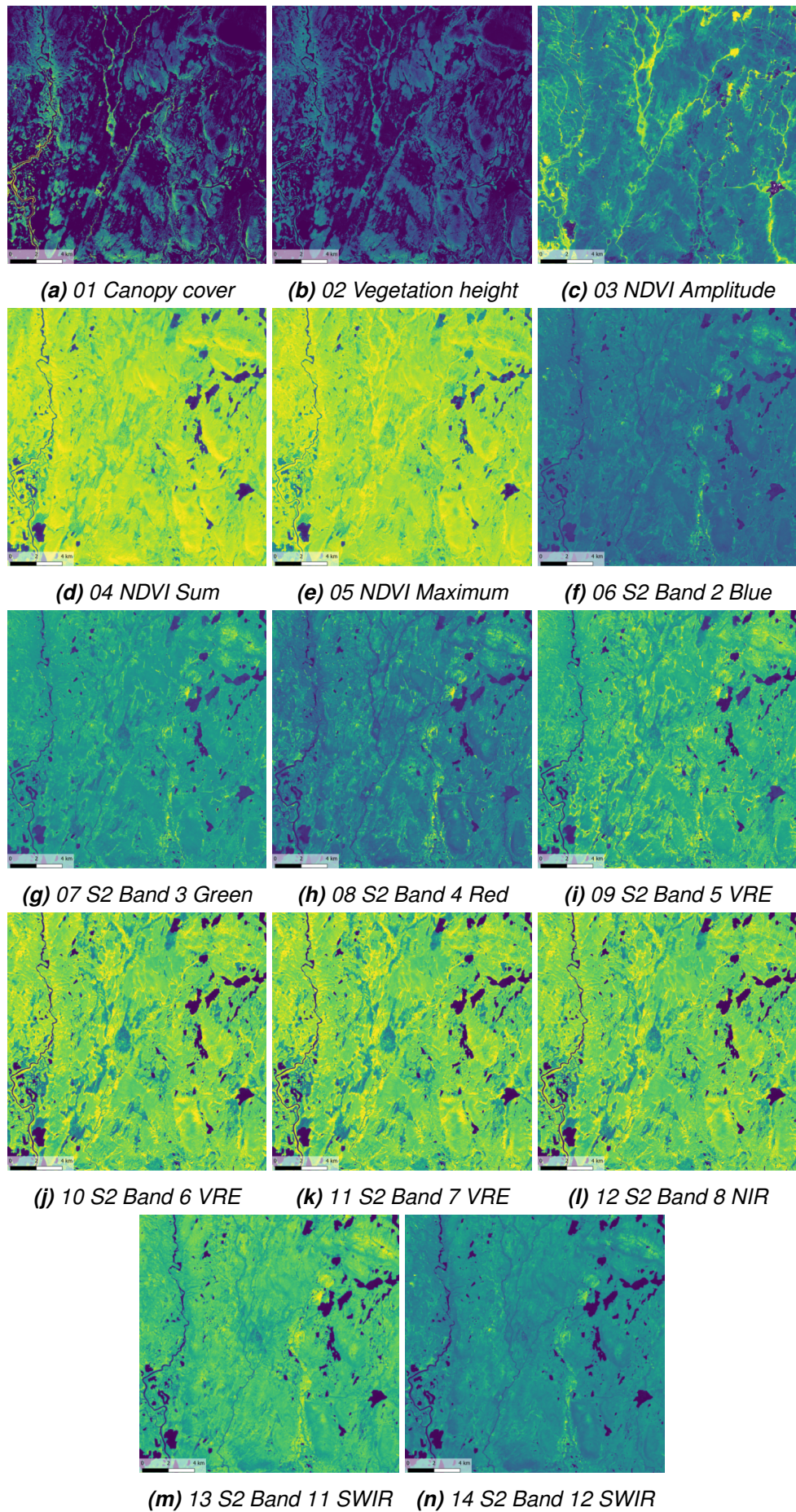


Figure A.2. Source raster channels separated for Lätäseno river area

GCS legend

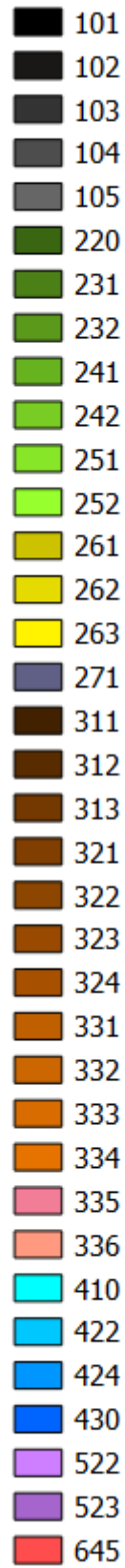


Figure A.3. GCS legends

Natura2000 legend


























	3110
	3160
	3220
	4060
	4080
	6150
	6270
	6430
	6450
	7140
	7160
	7220
	7230
	7240
	7310
	7320
	8110
	8210
	8220
	9010
	9040
	9050
	9080
	91D0
	91E0

Figure A.4. Natura2000 legends

APPENDIX B: FULL RESULT TABLES

name metric	RandomForest
AP macro	0.2477
AP micro	0.4140
AUC macro	0.7998
AUC micro	0.9051
F1 macro	0.2033
F1 weighted	0.3871
Prec. macro	0.2231
Prec. weighted	0.3775
Rec. macro	0.2094
Rec. weighted	0.4299
Top1 acc	0.4299
Top3 acc	0.7126
Top5 acc	0.8085

Table B.1. 5-fold cross-validated Random Forest results for GCS classes

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	ResNet	0.1896	0.2608	0.7348	0.8358	0.1742	0.3090	0.1846	0.3108	0.1868	0.3244	0.3244	0.5928	0.7148
	ResNet TTA	0.2016	0.2854	0.7598	0.8512	0.1890	0.3220	0.2028	0.3210	0.1984	0.3424	0.3424	0.6250	0.7356
Base crop	ResNet	0.1890	0.2762	0.7368	0.8630	0.1720	0.3156	0.1814	0.3116	0.1776	0.3344	0.3344	0.6138	0.7372
	ResNet TTA	0.2232	0.3304	0.7906	0.8890	0.1828	0.3382	0.2014	0.3348	0.1866	0.3686	0.3686	0.6622	0.7764
NS	ResNet	0.2032	0.2988	0.7732	0.8728	0.1838	0.3232	0.2104	0.3254	0.1882	0.3592	0.3592	0.6546	0.7690
	ResNet TTA	0.2126	0.3112	0.7822	0.8802	0.1804	0.3202	0.2004	0.3152	0.1856	0.3602	0.3602	0.6686	0.7796
NS crop	ResNet	0.1978	0.3056	0.7608	0.8748	0.1672	0.3232	0.1876	0.3178	0.1712	0.3576	0.3576	0.6518	0.7722
	ResNet TTA	0.2312	0.3478	0.7950	0.8946	0.1892	0.3492	0.2356	0.3572	0.1844	0.3934	0.3934	0.6860	0.7906
NS crop no freeze	ResNet	0.2172	0.2916	0.7712	0.8766	0.1914	0.3206	0.2094	0.3200	0.1978	0.3416	0.3416	0.6392	0.7598
	ResNet TTA	0.2298	0.3384	0.8014	0.8944	0.2016	0.3470	0.2384	0.3502	0.1984	0.3846	0.3846	0.6660	0.7852
NS no freeze	ResNet	0.1972	0.2914	0.7606	0.8630	0.1756	0.3196	0.1866	0.3116	0.1804	0.3428	0.3428	0.6160	0.7410
	ResNet TTA	0.2084	0.3038	0.7738	0.8722	0.1808	0.3244	0.1912	0.3152	0.1838	0.3484	0.3484	0.6390	0.7540
PT	ResNet	0.2048	0.3172	0.7822	0.8902	0.1710	0.3126	0.1838	0.3028	0.1766	0.3496	0.3496	0.6682	0.7730
	ResNet TTA	0.2144	0.3330	0.7850	0.8940	0.1810	0.3326	0.2046	0.3338	0.1842	0.3708	0.3708	0.6768	0.7882
PT crop	ResNet	0.2198	0.3276	0.8004	0.8962	0.1558	0.3012	0.1752	0.2932	0.1616	0.3474	0.3474	0.6754	0.7896
	ResNet TTA	0.2632	0.3612	0.8320	0.9064	0.1638	0.3144	0.2336	0.3552	0.1616	0.3784	0.3784	0.6936	0.8104
PT crop no freeze	ResNet	0.2102	0.2776	0.7702	0.8630	0.1860	0.3220	0.2044	0.3260	0.1886	0.3336	0.3336	0.6314	0.7576
	ResNet TTA	0.2424	0.3360	0.8006	0.8900	0.2058	0.3542	0.2310	0.3546	0.2024	0.3804	0.3804	0.6676	0.7880
PT no freeze	ResNet	0.1980	0.2796	0.7404	0.8470	0.1864	0.3344	0.1964	0.3294	0.1932	0.3494	0.3494	0.6192	0.7366
	ResNet TTA	0.1996	0.2912	0.7450	0.8538	0.1904	0.3358	0.2052	0.3310	0.1986	0.3538	0.3538	0.6278	0.7406
UPT	ResNet	0.1077	0.1960	0.6466	0.8334	0.0708	0.1814	0.0837	0.1714	0.0801	0.2456	0.2456	0.5116	0.6824
	ResNet TTA	0.1096	0.1966	0.6486	0.8342	0.0755	0.1880	0.0990	0.1942	0.0839	0.2504	0.2504	0.5170	0.6848

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	ResNet	0.0292	0.0258	0.0245	0.0200	0.0283	0.0243	0.0269	0.0211	0.0402	0.0270	0.0270	0.0175	0.0251
	ResNet TTA	0.0293	0.0196	0.0131	0.0154	0.0372	0.0270	0.0321	0.0232	0.0527	0.0247	0.0247	0.0110	0.0114
Base crop	ResNet	0.0135	0.0192	0.0242	0.0061	0.0244	0.0180	0.0283	0.0194	0.0301	0.0205	0.0205	0.0116	0.0078
	ResNet TTA	0.0129	0.0180	0.0068	0.0055	0.0183	0.0182	0.0221	0.0204	0.0231	0.0183	0.0183	0.0138	0.0201
NS	ResNet	0.0191	0.0275	0.0250	0.0093	0.0276	0.0168	0.0507	0.0282	0.0233	0.0118	0.0118	0.0136	0.0121
	ResNet TTA	0.0247	0.0291	0.0138	0.0091	0.0165	0.0104	0.0347	0.0207	0.0146	0.0123	0.0123	0.0140	0.0171
NS crop	ResNet	0.0049	0.0136	0.0129	0.0065	0.0228	0.0248	0.0403	0.0333	0.0223	0.0250	0.0250	0.0202	0.0181
	ResNet TTA	0.0219	0.0316	0.0208	0.0062	0.0242	0.0146	0.0479	0.0290	0.0217	0.0104	0.0104	0.0159	0.0136
NS crop no freeze	ResNet	0.0265	0.0335	0.0190	0.0088	0.0184	0.0214	0.0131	0.0172	0.0262	0.0197	0.0197	0.0166	0.0121
	ResNet TTA	0.0134	0.0245	0.0157	0.0072	0.0140	0.0061	0.0274	0.0064	0.0159	0.0067	0.0067	0.0163	0.0169
NS no freeze	ResNet	0.0213	0.0218	0.0215	0.0089	0.0230	0.0109	0.0345	0.0136	0.0197	0.0166	0.0166	0.0238	0.0220
	ResNet TTA	0.0199	0.0223	0.0161	0.0085	0.0260	0.0126	0.0410	0.0133	0.0199	0.0154	0.0154	0.0222	0.0232
PT	ResNet	0.0218	0.0299	0.0221	0.0083	0.0190	0.0102	0.0309	0.0183	0.0180	0.0115	0.0115	0.0261	0.0184
	ResNet TTA	0.0197	0.0269	0.0190	0.0070	0.0154	0.0171	0.0348	0.0419	0.0134	0.0211	0.0211	0.0166	0.0231
PT crop	ResNet	0.0083	0.0107	0.0252	0.0083	0.0203	0.0209	0.0162	0.0177	0.0189	0.0186	0.0186	0.0175	0.0176
	ResNet TTA	0.0090	0.0143	0.0175	0.0049	0.0206	0.0211	0.0356	0.0323	0.0194	0.0153	0.0153	0.0098	0.0139
PT crop no freeze	ResNet	0.0108	0.0176	0.0286	0.0107	0.0083	0.0178	0.0175	0.0183	0.0130	0.0143	0.0143	0.0257	0.0276
	ResNet TTA	0.0109	0.0202	0.0289	0.0089	0.0127	0.0232	0.0214	0.0203	0.0124	0.0200	0.0200	0.0193	0.0126
PT no freeze	ResNet	0.0203	0.0166	0.0214	0.0142	0.0150	0.0240	0.0198	0.0290	0.0158	0.0213	0.0213	0.0285	0.0267
	ResNet TTA	0.0133	0.0165	0.0159	0.0109	0.0150	0.0258	0.0201	0.0276	0.0131	0.0247	0.0247	0.0153	0.0170
UPT	ResNet	0.0116	0.0095	0.0285	0.0083	0.0122	0.0096	0.0161	0.0178	0.0084	0.0079	0.0079	0.0267	0.0160
	ResNet TTA	0.0113	0.0084	0.0270	0.0079	0.0104	0.0112	0.0202	0.0371	0.0085	0.0102	0.0102	0.0263	0.0189

Table B.2. 5-fold cross-validated ResNet18 results and standard deviations for GCS classes

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	Ens.	0.2564	0.3852	0.8364	0.9120	0.1810	0.3240	0.1936	0.3236	0.1932	0.3450	0.3450	0.7060	0.8054
	Ens. TTA	0.2640	0.3976	0.8314	0.9122	0.1874	0.3358	0.2076	0.3348	0.1948	0.3616	0.3616	0.7114	0.8142
Base crop	Ens.	0.2504	0.3896	0.8288	0.9110	0.1840	0.3376	0.1948	0.3302	0.1892	0.3630	0.3630	0.7076	0.8046
	Ens. TTA	0.2734	0.4184	0.8374	0.9158	0.2048	0.3690	0.2222	0.3620	0.2064	0.4058	0.4058	0.7158	0.8118
NS	Ens.	0.2600	0.4064	0.8392	0.9146	0.1974	0.3600	0.2182	0.3526	0.2022	0.4052	0.4052	0.7096	0.8164
	Ens. TTA	0.2746	0.4120	0.8360	0.9134	0.1998	0.3638	0.2214	0.3564	0.2068	0.4106	0.4106	0.7264	0.8128
NS crop	Ens.	0.2564	0.4142	0.8286	0.9128	0.1988	0.3710	0.2254	0.3668	0.2016	0.4136	0.4136	0.7190	0.8108
	Ens. TTA	0.2732	0.4292	0.8406	0.9170	0.2030	0.3792	0.2502	0.3818	0.2012	0.4302	0.4302	0.7234	0.8204
NS crop no freeze	Ens.	0.2668	0.3996	0.8396	0.9152	0.2108	0.3536	0.2342	0.3504	0.2130	0.3858	0.3858	0.7168	0.8206
	Ens. TTA	0.2808	0.4262	0.8418	0.9168	0.2076	0.3790	0.2354	0.3790	0.2062	0.4222	0.4222	0.7326	0.8176
NS no freeze	Ens.	0.2532	0.3992	0.8240	0.9120	0.1842	0.3498	0.2042	0.3412	0.1880	0.3812	0.3812	0.7066	0.8088
	Ens. TTA	0.2580	0.3980	0.8236	0.9120	0.1956	0.3500	0.2110	0.3368	0.2014	0.3848	0.3848	0.7036	0.8108
PT	Ens.	0.2608	0.4172	0.8340	0.9154	0.1956	0.3698	0.2168	0.3618	0.2028	0.4206	0.4206	0.7302	0.8210
	Ens. TTA	0.2732	0.4220	0.8244	0.9160	0.2010	0.3766	0.2316	0.3742	0.2060	0.4266	0.4266	0.7288	0.8136
PT crop	Ens.	0.2700	0.4216	0.8380	0.9172	0.1902	0.3674	0.2132	0.3648	0.1966	0.4196	0.4196	0.7326	0.8224
	Ens. TTA	0.2922	0.4342	0.8530	0.9202	0.1968	0.3704	0.2444	0.3846	0.1974	0.4260	0.4260	0.7282	0.8224
PT crop no freeze	Ens.	0.2646	0.3918	0.8296	0.9134	0.1982	0.3388	0.2210	0.3408	0.1980	0.3560	0.3560	0.7152	0.8150
	Ens. TTA	0.2782	0.4212	0.8368	0.9168	0.2110	0.3700	0.2378	0.3698	0.2096	0.4036	0.4036	0.7242	0.8196
PT no freeze	Ens.	0.2590	0.3968	0.8270	0.9136	0.1854	0.3398	0.1954	0.3320	0.1934	0.3598	0.3598	0.7158	0.8122
	Ens. TTA	0.2530	0.4006	0.8240	0.9136	0.1966	0.3528	0.2090	0.3444	0.2042	0.3756	0.3756	0.7190	0.8154
UPT	Ens.	0.2414	0.3908	0.7882	0.8974	0.1788	0.3486	0.2120	0.3510	0.1822	0.4102	0.4102	0.6768	0.7806
	Ens. TTA	0.2436	0.3886	0.7844	0.8982	0.1746	0.3428	0.2054	0.3472	0.1782	0.4036	0.4036	0.6768	0.7880

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	Ens.	0.0179	0.0216	0.0187	0.0057	0.0221	0.0205	0.0149	0.0183	0.0348	0.0230	0.0230	0.0212	0.0202
	Ens. TTA	0.0216	0.0212	0.0204	0.0055	0.0256	0.0154	0.0307	0.0136	0.0355	0.0141	0.0141	0.0171	0.0158
Base crop	Ens.	0.0232	0.0292	0.0050	0.0032	0.0277	0.0319	0.0330	0.0310	0.0328	0.0394	0.0394	0.0130	0.0162
	Ens. TTA	0.0152	0.0251	0.0147	0.0039	0.0192	0.0194	0.0277	0.0191	0.0200	0.0218	0.0218	0.0088	0.0132
NS	Ens.	0.0283	0.0246	0.0098	0.0046	0.0214	0.0159	0.0303	0.0227	0.0210	0.0136	0.0136	0.0280	0.0136
	Ens. TTA	0.0351	0.0248	0.0103	0.0048	0.0160	0.0105	0.0207	0.0169	0.0133	0.0109	0.0109	0.0155	0.0167
NS crop	Ens.	0.0157	0.0187	0.0110	0.0034	0.0190	0.0223	0.0410	0.0340	0.0150	0.0250	0.0250	0.0176	0.0208
	Ens. TTA	0.0289	0.0225	0.0194	0.0051	0.0097	0.0075	0.0409	0.0210	0.0070	0.0067	0.0067	0.0196	0.0215
NS crop no freeze	Ens.	0.0168	0.0278	0.0093	0.0036	0.0192	0.0131	0.0264	0.0123	0.0236	0.0182	0.0182	0.0156	0.0239
	Ens. TTA	0.0262	0.0202	0.0211	0.0031	0.0166	0.0081	0.0125	0.0060	0.0189	0.0059	0.0059	0.0175	0.0182
NS no freeze	Ens.	0.0252	0.0173	0.0110	0.0045	0.0125	0.0202	0.0281	0.0228	0.0117	0.0194	0.0194	0.0294	0.0329
	Ens. TTA	0.0119	0.0172	0.0190	0.0032	0.0243	0.0128	0.0346	0.0105	0.0221	0.0186	0.0186	0.0270	0.0252
PT	Ens.	0.0150	0.0220	0.0283	0.0052	0.0100	0.0131	0.0292	0.0236	0.0080	0.0123	0.0123	0.0121	0.0211
	Ens. TTA	0.0196	0.0205	0.0281	0.0040	0.0054	0.0113	0.0258	0.0144	0.0041	0.0156	0.0156	0.0173	0.0179
PT crop	Ens.	0.0188	0.0191	0.0205	0.0043	0.0100	0.0097	0.0174	0.0141	0.0078	0.0084	0.0084	0.0091	0.0175
	Ens. TTA	0.0200	0.0158	0.0149	0.0031	0.0112	0.0099	0.0222	0.0120	0.0073	0.0070	0.0070	0.0144	0.0214
PT crop no freeze	Ens.	0.0120	0.0211	0.0181	0.0042	0.0128	0.0191	0.0172	0.0205	0.0146	0.0151	0.0151	0.0172	0.0209
	Ens. TTA	0.0133	0.0244	0.0241	0.0047	0.0133	0.0250	0.0138	0.0226	0.0134	0.0208	0.0208	0.0145	0.0227
PT no freeze	Ens.	0.0108	0.0185	0.0132	0.0057	0.0092	0.0226	0.0131	0.0253	0.0129	0.0197	0.0197	0.0245	0.0218
	Ens. TTA	0.0138	0.0203	0.0147	0.0045	0.0180	0.0282	0.0256	0.0305	0.0187	0.0264	0.0264	0.0265	0.0188
UPT	Ens.	0.0147	0.0120	0.0252	0.0027	0.0072	0.0147	0.0137	0.0201	0.0048	0.0178	0.0178	0.0218	0.0178
	Ens. TTA	0.0155	0.0121	0.0238	0.0037	0.0137	0.0122	0.0286	0.0253	0.0091	0.0128	0.0128	0.0202	0.0206

Table B.3. 5-fold cross-validated ResNet18+RandomForest ensemble model results and standard deviations for GCS classes

metric model	F1 weighted	Prec. weighted	Rec. weighted/Acc	Top3 acc
Base	0.322	0.321	0.342	0.625
Base crop	0.338	0.335	0.369	0.662
NS	0.320	0.315	0.360	0.669
NS crop	0.349	0.357	0.393	0.686
NS crop no freeze	0.347	0.350	0.385	0.666
NS no freeze	0.324	0.315	0.348	0.639
PT	0.333	0.334	0.371	0.677
PT crop	0.314	0.355	0.378	0.694
PT crop no freeze	0.354	0.355	0.380	0.668
PT no freeze	0.336	0.331	0.354	0.628
UPT	0.188	0.194	0.250	0.517
RandomForest	0.387	0.378	0.430	0.713

Table B.4. ResNet results for GCS classes: Selected metrics for GCS classification for different ResNet models and the random forest baseline, with test-time augmentation. The results in this table are a subset of Table B.2. Refer to Table 7.1 for abbreviations.

metric model	F1 weighted	Prec. weighted	Rec. weighted/Acc	Top3 acc
Base	0.336	0.335	0.362	0.711
Base crop	0.369	0.362	0.406	0.716
NS	0.364	0.356	0.411	0.726
NS crop	0.379	0.382	0.430	0.723
NS crop no freeze	0.379	0.379	0.422	0.733
NS no freeze	0.350	0.337	0.385	0.704
PT	0.377	0.374	0.427	0.729
PT crop	0.370	0.385	0.426	0.728
PT crop no freeze	0.370	0.370	0.404	0.724
PT no freeze	0.353	0.344	0.376	0.719
UPT	0.343	0.347	0.404	0.677
RandomForest	0.387	0.378	0.430	0.713

Table B.5. Ensemble results for GCS classes: Selected metrics for GCS classification for different ensemble models and the random forest baseline, with test-time augmentation. The results in this table are a subset of Table B.3. Refer to Table 7.1 for abbreviations.

name metric	RandomForest
AP macro	0.3242
AP micro	0.6319
AUC macro	0.8228
AUC micro	0.9336
F1 macro	0.2688
F1 weighted	0.5389
Prec. macro	0.3107
Prec. weighted	0.5331
Rec. macro	0.2653
Rec. weighted	0.5791
Top1 acc	0.5791
Top3 acc	0.8133
Top5 acc	0.8938

Table B.6. 5-fold cross-validated Random Forest results for Natura2000 classes

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	ResNet	0.2876	0.4990	0.7900	0.8840	0.2564	0.4734	0.2650	0.4658	0.2638	0.4930	0.4930	0.7284	0.8212
	ResNet TTA	0.2934	0.5166	0.8022	0.8956	0.2522	0.4806	0.2636	0.4690	0.2598	0.5092	0.5092	0.7508	0.8420
Base crop	ResNet	0.2786	0.4852	0.8012	0.9000	0.2472	0.4712	0.2692	0.4652	0.2484	0.4956	0.4956	0.7540	0.8522
	ResNet TTA	0.3098	0.5486	0.8352	0.9228	0.2610	0.4908	0.2990	0.4788	0.2552	0.5356	0.5356	0.7910	0.8850
NS	ResNet	0.3110	0.4984	0.7962	0.9056	0.2604	0.4860	0.3096	0.4898	0.2562	0.5216	0.5216	0.7800	0.8774
	ResNet TTA	0.3236	0.5210	0.7990	0.9116	0.2688	0.5024	0.3240	0.5004	0.2618	0.5412	0.5412	0.7952	0.8792
NS crop	ResNet	0.3096	0.5078	0.7774	0.9058	0.2684	0.4878	0.3216	0.4884	0.2630	0.5190	0.5190	0.7788	0.8676
	ResNet TTA	0.3222	0.5560	0.8020	0.9222	0.2648	0.5070	0.3286	0.5052	0.2572	0.5538	0.5538	0.8004	0.8848
NS crop no freeze	ResNet	0.3012	0.5138	0.7836	0.9078	0.2650	0.4846	0.2986	0.4814	0.2622	0.5070	0.5070	0.7654	0.8596
	ResNet TTA	0.3136	0.4996	0.8200	0.9104	0.1654	0.4532	0.2124	0.4564	0.1740	0.5154	0.5154	0.7476	0.8528
NS no freeze	ResNet	0.3008	0.5292	0.8040	0.9130	0.2720	0.4900	0.3020	0.4778	0.2672	0.5182	0.5182	0.7662	0.8784
	ResNet TTA	0.3092	0.5492	0.8210	0.9198	0.2656	0.4934	0.2842	0.4780	0.2702	0.5256	0.5256	0.7802	0.8844
PT	ResNet	0.3140	0.5626	0.8156	0.9248	0.2734	0.5048	0.3116	0.4998	0.2722	0.5400	0.5400	0.7894	0.8780
	ResNet TTA	0.3210	0.5794	0.8226	0.9270	0.2890	0.5144	0.3370	0.5168	0.2864	0.5500	0.5500	0.7944	0.8836
PT crop	ResNet	0.3292	0.5566	0.8106	0.9254	0.2354	0.4788	0.2858	0.4770	0.2318	0.5232	0.5232	0.7822	0.8858
	ResNet TTA	0.3502	0.5954	0.8366	0.9338	0.2224	0.4928	0.2764	0.4946	0.2224	0.5538	0.5538	0.8098	0.9030
PT crop no freeze	ResNet	0.2818	0.4914	0.7864	0.8960	0.2514	0.4704	0.2616	0.4662	0.2534	0.4856	0.4856	0.7562	0.8552
	ResNet TTA	0.3066	0.5544	0.8092	0.9180	0.2742	0.5040	0.3144	0.4982	0.2660	0.5370	0.5370	0.7844	0.8780
PT no freeze	ResNet	0.2916	0.4936	0.8040	0.8946	0.2672	0.4696	0.2872	0.4606	0.2676	0.4876	0.4876	0.7576	0.8568
	ResNet TTA	0.3128	0.5144	0.8174	0.9014	0.2798	0.4856	0.2954	0.4748	0.2804	0.5078	0.5078	0.7754	0.8642
UPT	ResNet	0.1794	0.3998	0.7246	0.8872	0.1338	0.3636	0.1274	0.3184	0.1500	0.4350	0.4350	0.7142	0.8196
	ResNet TTA	0.1764	0.3992	0.7164	0.8868	0.1350	0.3658	0.1299	0.3210	0.1508	0.4386	0.4386	0.7134	0.8206

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	ResNet	0.0289	0.0142	0.0218	0.0117	0.0247	0.0127	0.0287	0.0108	0.0315	0.0116	0.0116	0.0158	0.0253
	ResNet TTA	0.0286	0.0124	0.0275	0.0133	0.0224	0.0133	0.0259	0.0146	0.0343	0.0127	0.0127	0.0152	0.0150
Base crop	ResNet	0.0223	0.0237	0.0322	0.0120	0.0314	0.0167	0.0391	0.0160	0.0312	0.0188	0.0188	0.0185	0.0148
	ResNet TTA	0.0156	0.0232	0.0277	0.0108	0.0351	0.0243	0.0448	0.0299	0.0319	0.0209	0.0209	0.0121	0.0171
NS	ResNet	0.0388	0.0277	0.0334	0.0121	0.0325	0.0314	0.0374	0.0449	0.0309	0.0303	0.0303	0.0270	0.0163
	ResNet TTA	0.0318	0.0356	0.0315	0.0130	0.0193	0.0256	0.0275	0.0303	0.0193	0.0291	0.0291	0.0166	0.0119
NS crop	ResNet	0.0165	0.0325	0.0351	0.0098	0.0198	0.0176	0.0490	0.0251	0.0111	0.0255	0.0255	0.0169	0.0074
	ResNet TTA	0.0190	0.0366	0.0328	0.0095	0.0129	0.0132	0.0440	0.0194	0.0143	0.0141	0.0141	0.0115	0.0083
NS crop no freeze	ResNet	0.0329	0.0349	0.0307	0.0093	0.0369	0.0171	0.0531	0.0218	0.0360	0.0173	0.0173	0.0086	0.0070
	ResNet TTA	0.0284	0.0510	0.0230	0.0087	0.0132	0.0193	0.0085	0.0129	0.0142	0.0235	0.0235	0.0141	0.0091
NS no freeze	ResNet	0.0284	0.0259	0.0265	0.0070	0.0178	0.0171	0.0289	0.0202	0.0174	0.0191	0.0191	0.0161	0.0122
	ResNet TTA	0.0345	0.0326	0.0204	0.0081	0.0264	0.0163	0.0334	0.0171	0.0325	0.0186	0.0186	0.0165	0.0160
PT	ResNet	0.0324	0.0350	0.0336	0.0078	0.0311	0.0136	0.0238	0.0099	0.0319	0.0172	0.0172	0.0128	0.0092
	ResNet TTA	0.0342	0.0302	0.0241	0.0060	0.0354	0.0144	0.0256	0.0224	0.0397	0.0201	0.0201	0.0155	0.0105
PT crop	ResNet	0.0261	0.0204	0.0171	0.0040	0.0101	0.0067	0.0345	0.0166	0.0108	0.0076	0.0076	0.0132	0.0102
	ResNet TTA	0.0106	0.0153	0.0321	0.0045	0.0224	0.0198	0.0286	0.0256	0.0268	0.0161	0.0161	0.0100	0.0082
PT crop no freeze	ResNet	0.0239	0.0278	0.0269	0.0112	0.0118	0.0142	0.0139	0.0129	0.0152	0.0159	0.0159	0.0213	0.0150
	ResNet TTA	0.0228	0.0302	0.0239	0.0121	0.0299	0.0053	0.0447	0.0054	0.0314	0.0052	0.0052	0.0140	0.0191
PT no freeze	ResNet	0.0276	0.0388	0.0324	0.0121	0.0319	0.0286	0.0434	0.0327	0.0309	0.0259	0.0259	0.0118	0.0287
	ResNet TTA	0.0412	0.0320	0.0324	0.0118	0.0342	0.0225	0.0418	0.0289	0.0355	0.0191	0.0191	0.0226	0.0170
UPT	ResNet	0.0224	0.0098	0.0270	0.0061	0.0218	0.0143	0.0238	0.0162	0.0201	0.0155	0.0155	0.0177	0.0110
	ResNet TTA	0.0194	0.0101	0.0198	0.0061	0.0210	0.0134	0.0213	0.0137	0.0197	0.0144	0.0144	0.0226	0.0110

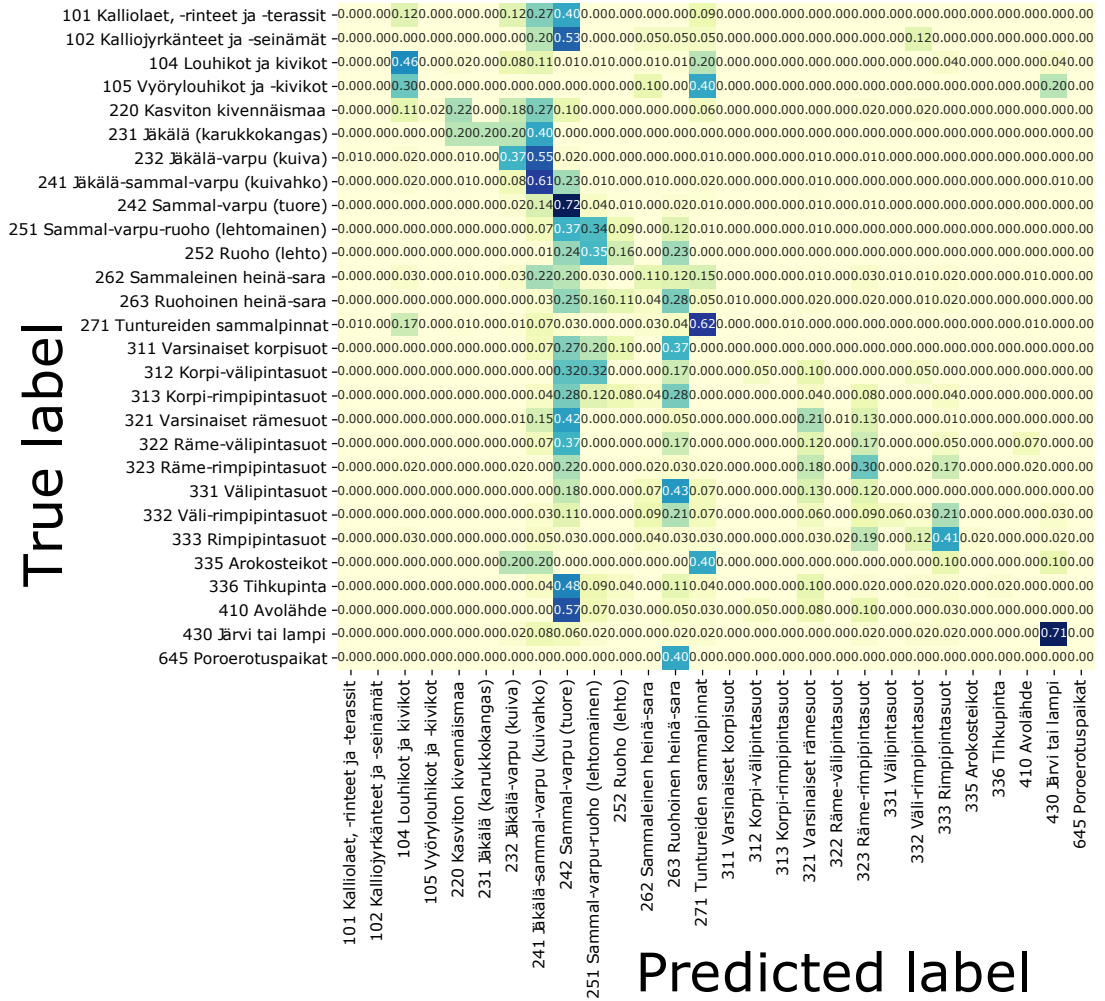
Table B.7. 5-fold cross-validated ResNet18 results and standard deviations for Natura2000 classes

model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	Ens.	0.3398	0.6160	0.8450	0.9356	0.2684	0.4922	0.2794	0.4820	0.2772	0.5178	0.5178	0.8104	0.8936
	Ens. TTA	0.3330	0.6200	0.8486	0.9366	0.2626	0.4944	0.2716	0.4784	0.2688	0.5272	0.5272	0.8052	0.8914
Base crop	Ens.	0.3304	0.6176	0.8490	0.9356	0.2666	0.4970	0.2952	0.4904	0.2654	0.5270	0.5270	0.8122	0.8952
	Ens. TTA	0.3444	0.6340	0.8670	0.9410	0.2632	0.5036	0.3046	0.4936	0.2586	0.5534	0.5534	0.8226	0.9036
NS	Ens.	0.3516	0.6298	0.8602	0.9398	0.2704	0.5122	0.3286	0.5142	0.2666	0.5542	0.5542	0.8204	0.9032
	Ens. TTA	0.3642	0.6346	0.8562	0.9402	0.2714	0.5192	0.3302	0.5122	0.2650	0.5652	0.5652	0.8200	0.8966
NS crop	Ens.	0.3546	0.6344	0.8392	0.9382	0.2858	0.5308	0.3366	0.5260	0.2818	0.5732	0.5732	0.8248	0.9054
	Ens. TTA	0.3532	0.6452	0.8620	0.9422	0.2672	0.5336	0.3208	0.5248	0.2632	0.5860	0.5860	0.8224	0.9076
NS crop no freeze	Ens.	0.3370	0.6242	0.8434	0.9376	0.2858	0.5208	0.3276	0.5150	0.2812	0.5540	0.5540	0.8138	0.8944
	Ens. TTA	0.3406	0.6312	0.8520	0.9372	0.2386	0.5216	0.2866	0.5206	0.2428	0.5850	0.5850	0.8116	0.8956
NS no freeze	Ens.	0.3412	0.6308	0.8484	0.9394	0.2720	0.5116	0.3068	0.4994	0.2700	0.5486	0.5486	0.8146	0.9016
	Ens. TTA	0.3408	0.6328	0.8528	0.9408	0.2758	0.5172	0.3036	0.5036	0.2772	0.5548	0.5548	0.8160	0.9028
PT	Ens.	0.3562	0.6490	0.8490	0.9416	0.2898	0.5488	0.3512	0.5580	0.2842	0.5964	0.5964	0.8218	0.9042
	Ens. TTA	0.3504	0.6524	0.8588	0.9430	0.2870	0.5434	0.3506	0.5504	0.2814	0.5904	0.5904	0.8196	0.9020
PT crop	Ens.	0.3608	0.6454	0.8556	0.9424	0.2664	0.5332	0.3470	0.5508	0.2604	0.5806	0.5806	0.8272	0.9046
	Ens. TTA	0.3662	0.6550	0.8656	0.9434	0.2564	0.5324	0.3404	0.5452	0.2516	0.5890	0.5890	0.8306	0.9048
PT crop no freeze	Ens.	0.3258	0.6096	0.8348	0.9358	0.2708	0.4946	0.2842	0.4848	0.2724	0.5176	0.5176	0.8192	0.8988
	Ens. TTA	0.3422	0.6340	0.8524	0.9392	0.2804	0.5120	0.3246	0.5112	0.2750	0.5508	0.5508	0.8230	0.9042
PT no freeze	Ens.	0.3382	0.6098	0.8616	0.9392	0.2694	0.4832	0.2914	0.4714	0.2714	0.5080	0.5080	0.8166	0.9010
	Ens. TTA	0.3434	0.6174	0.8644	0.9402	0.2796	0.4986	0.3010	0.4852	0.2782	0.5256	0.5256	0.8232	0.9026
UPT	Ens.	0.3136	0.6000	0.8398	0.9298	0.2064	0.4780	0.2466	0.4578	0.2182	0.5588	0.5588	0.7806	0.8784
	Ens. TTA	0.3174	0.5990	0.8322	0.9296	0.2106	0.4844	0.2622	0.4844	0.2208	0.5628	0.5628	0.7942	0.8810

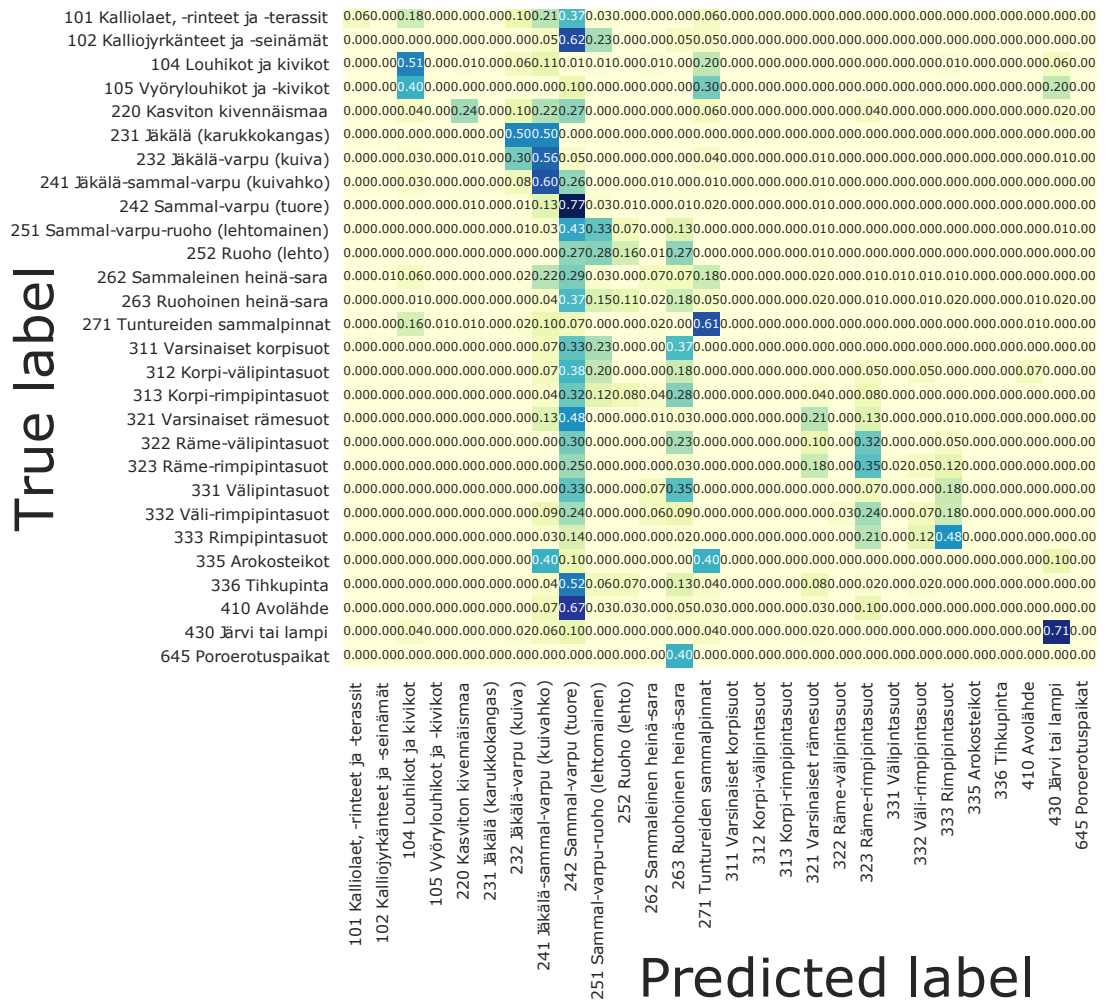
model	metric name	AP macro	AP micro	AUC macro	AUC micro	F1 macro	F1 weighted	Prec. macro	Prec. weighted	Rec. macro	Rec. weighted	Top1 acc	Top3 acc	Top5 acc
Base	Ens.	0.0193	0.0137	0.0152	0.0061	0.0267	0.0138	0.0285	0.0143	0.0315	0.0114	0.0114	0.0176	0.0095
	Ens. TTA	0.0206	0.0123	0.0248	0.0073	0.0281	0.0086	0.0246	0.0063	0.0306	0.0104	0.0104	0.0034	0.0088
Base crop	Ens.	0.0214	0.0184	0.0210	0.0089	0.0279	0.0178	0.0344	0.0155	0.0266	0.0203	0.0203	0.0101	0.0105
	Ens. TTA	0.0190	0.0180	0.0217	0.0078	0.0277	0.0139	0.0350	0.0128	0.0243	0.0102	0.0102	0.0126	0.0212
NS	Ens.	0.0253	0.0203	0.0280	0.0055	0.0217	0.0100	0.0226	0.0215	0.0163	0.0149	0.0149	0.0180	0.0113
	Ens. TTA	0.0297	0.0223	0.0214	0.0064	0.0289	0.0168	0.0498	0.0264	0.0210	0.0184	0.0184	0.0195	0.0089
NS crop	Ens.	0.0142	0.0205	0.0399	0.0082	0.0206	0.0073	0.0536	0.0177	0.0083	0.0157	0.0157	0.0078	0.0149
	Ens. TTA	0.0127	0.0204	0.0193	0.0061	0.0159	0.0135	0.0242	0.0209	0.0164	0.0160	0.0160	0.0098	0.0080
NS crop no freeze	Ens.	0.0249	0.0181	0.0208	0.0081	0.0271	0.0126	0.0408	0.0145	0.0264	0.0109	0.0109	0.0108	0.0151
	Ens. TTA	0.0170	0.0206	0.0093	0.0032	0.0271	0.0264	0.0355	0.0282	0.0239	0.0185	0.0185	0.0073	0.0053
NS no freeze	Ens.	0.0234	0.0149	0.0135	0.0061	0.0172	0.0126	0.0174	0.0185	0.0170	0.0129	0.0129	0.0025	0.0132
	Ens. TTA	0.0198	0.0192	0.0325	0.0072	0.0269	0.0075	0.0316	0.0127	0.0252	0.0109	0.0109	0.0092	0.0078
PT	Ens.	0.0275	0.0201	0.0304	0.0058	0.0318	0.0107	0.0436	0.0395	0.0262	0.0077	0.0077	0.0101	0.0123
	Ens. TTA	0.0174	0.0209	0.0190	0.0060	0.0194	0.0127	0.0427	0.0442	0.0197	0.0130	0.0130	0.0110	0.0150
PT crop	Ens.	0.0172	0.0141	0.0197	0.0042	0.0116	0.0070	0.0400	0.0128	0.0074	0.0056	0.0056	0.0090	0.0099
	Ens. TTA	0.0178	0.0130	0.0173	0.0044	0.0124	0.0139	0.0437	0.0193	0.0128	0.0097	0.0097	0.0083	0.0080
PT crop no freeze	Ens.	0.0124	0.0173	0.0254	0.0073	0.0165	0.0143	0.0221	0.0143	0.0171	0.0133	0.0133	0.0173	0.0129
	Ens. TTA	0.0176	0.0154	0.0216	0.0081	0.0296	0.0113	0.0508	0.0165	0.0283	0.0104	0.0104	0.0063	0.0102
PT no freeze	Ens.	0.0272	0.0273	0.0210	0.0078	0.0304	0.0255	0.0337	0.0286	0.0304	0.0211	0.0211	0.0075	0.0130
	Ens. TTA	0.0279	0.0203	0.0162	0.0065	0.0304	0.0180	0.0469	0.0237	0.0254	0.0123	0.0123	0.0067	0.0108
UPT	Ens.	0.0166	0.0103	0.0101	0.0036	0.0168	0.0171	0.0660	0.0533	0.0122	0.0107	0.0107	0.0098	0.0096
	Ens. TTA	0.0188	0.0106	0.0178	0.0049	0.0084	0.0116	0.0315	0.0223	0.0079	0.0112	0.0112	0.0051	0.0072

Table B.8. 5-fold cross-validated ResNet18+RandomForest ensemble model results and standard deviations for Natura2000 classes

APPENDIX C: ADDITIONAL FIGURES

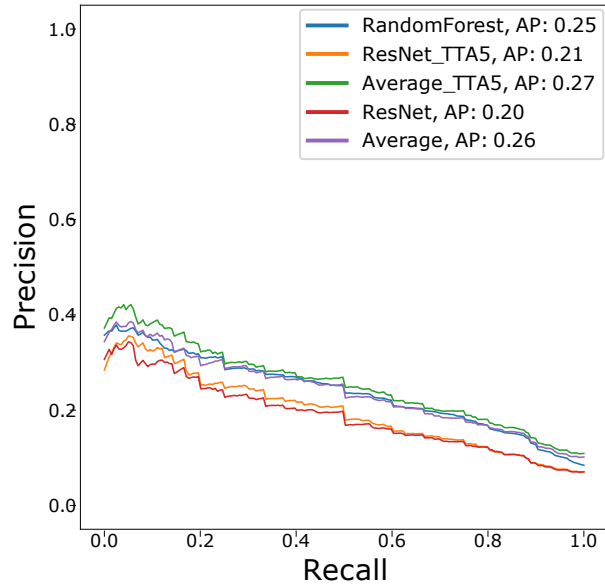


(a) Random forest

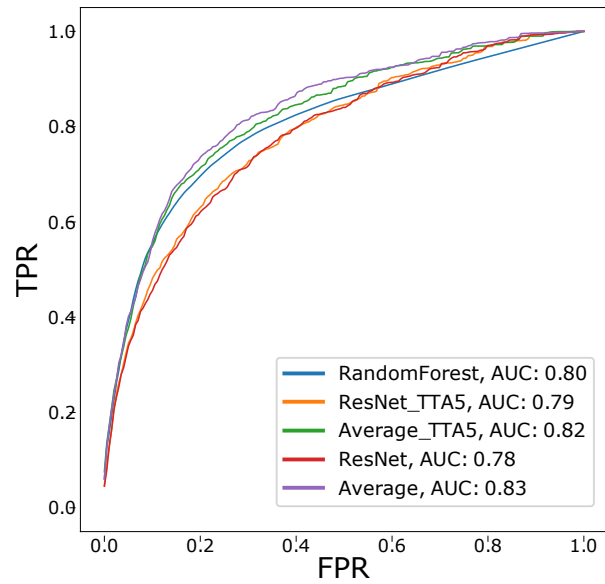


(c) Ensemble

Figure C.1. Normalized confusion matrices for GCS classes using CORINE-pretrained models with test-time augmentation applied

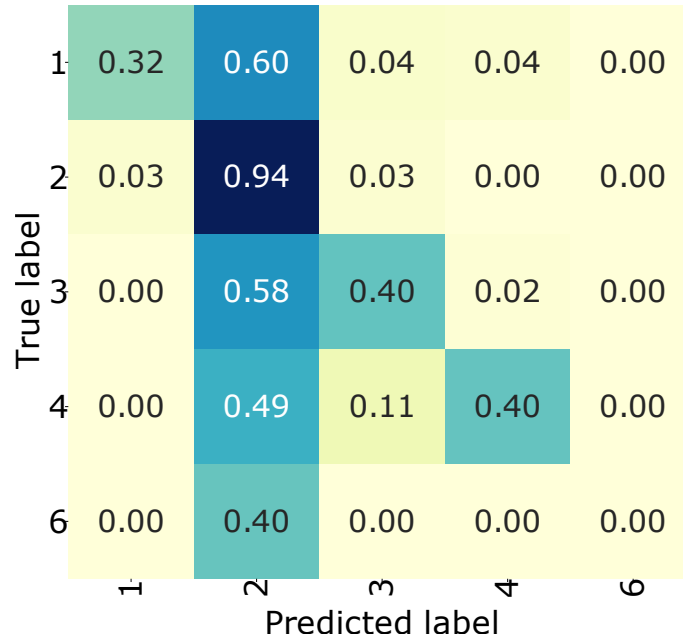
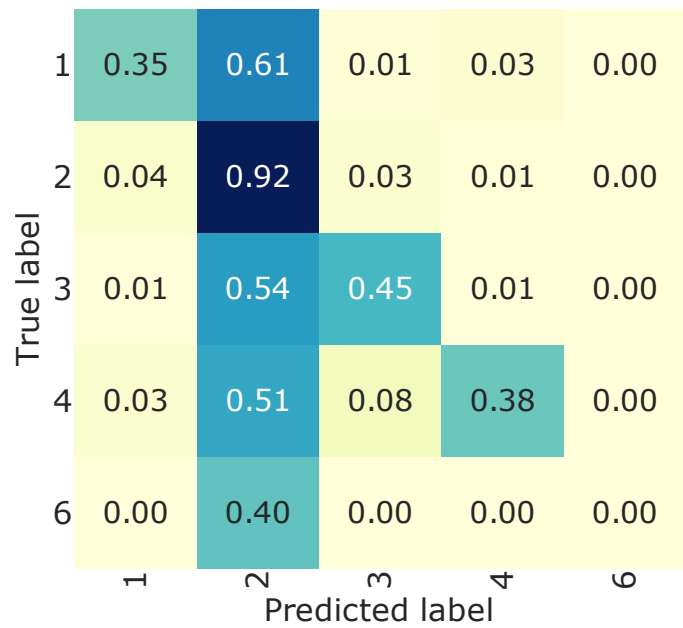


(a) Macro average precision-recall curves



(b) Macro average ROC curves

Figure C.2. GCS classes precision-recall and ROC curve comparisons between CORINE-pretrained ResNet, random forest, and ensemble models with test-time augmentation applied five times (TTA5)

(a) *Random forest*(b) *ResNet***Figure C.3.** Highest hierarchy level normalized confusion matrices for GCS classes