Master Computer Science

Systematic Comparison of Few-Shot Learners in Remote Sensing Image Recognition Tasks

Name:	Gareth Kok
Student ID:	s2989808
Date:	26/08/2024
Specialisation:	Data Science
1st supervisor:	Mitra Baratchi
2nd supervisor:	Jan van Rijn

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Acknowledgement

This work was performed using the LIACS compute resources provide by Leiden University. Special thanks to Mitra Baratchi and Jan van Rijn for their patience and guidance throughout the process.

Abstract

Remote sensing image scene classification has provided researchers with a wealth of data for performing tasks from mapping and monitoring to planning and response. When developing solutions for the abovementioned practical applications of remote sensing imagery data, machine learning models are generally used and continue to be developed to improve on previous approaches and create new use cases. The success of deep learning in domains outside of remote sensing (i.e., natural scene imagery) has led researchers to apply the same or similar techniques to various remote sensing tasks. The success of these models is in large part due to the availability of large labelled datasets, which in the case of remote sensing is not always practical to obtain hence researchers have explored the use of few-shot learning techniques. In this thesis, we aim to compare the performance of several well-established few-shot learning models in the remote sensing domain, focusing on how these models perform under different real-world settings providing a comprehensive overview of model performance. Few-shot learning alleviates several bottlenecks faced by traditional supervised learning methods, particularly the reliance on large labelled datasets. The research evaluates several few-shot learning approaches, including transfer learning (Baseline and Baseline++) and meta-learning (MAML, ProtoNet, RelationNet, MatchingNet) based methods, across 5 diverse remote sensing image classification datasets. The experiments carried out include an assessment of how the models perform in a standard few-shot learning setting followed by more comprehensive experiments, by varying the network depth, cross-dataset training and testing, and the injection of class imbalance. The results provide insights into the effectiveness of the few-shot learning models in different scenarios within the remote sensing domain, with the Baseline transfer learning method and Prototypical Network Meta-learning method both heavily outperforming the other methods in the network depth and class imbalance settings, while the meta-learning methods proving to be effective in a low-shot setting (1-shot). In general, based on the research conducted it is suggested to use the Baseline transfer learning method when the training dataset is large and diverse enough to do so or the standard ProtoNet architecture when the training data is limited.

Contents

1 Introduction

With practical applications spanning from mapping and monitoring to planning and response, remote sensing provides researchers with a global perspective and a wealth of data. The data contributes significantly to the decision-making processes which remote sensing practitioners typically deal with. Remote sensing refers to the acquisition of information from a distance, with a common source of information being imagery from satellites, aircraft, and drones. When developing solutions for the above-mentioned practical applications of remote sensing data, machine learning models are generally used and continue to be developed to improve on previous approaches and create new use cases. Specifically, deep learning is emerging as the go-to method for various computer vision tasks producing cutting-edge results [34, 42].

The effectiveness of deep learning in domains beyond remote sensing, such as natural scene imagery, has encouraged researchers to apply similar techniques to a range of remote sensing tasks, including segmentation, object detection, and classification. Major strides have been made thanks to model-driven development, open-source research communities, and an increase in affordable hardware [21]. That said, the success of deep learning is largely attributed to the availability of extensive labeled datasets, which, in the context of remote sensing, are not always feasible to obtain. Data collection in the domain of remote sensing is vast, however, the cost and effort involved in labelling said data is a bottleneck when practitioners want to perform predictions in an accurate and timely manner [26]. Simply put, with the large volumes of remote sensing data being produced and with there being enough computing power to cope with the volumes of data [21], the major issue is a lack of ground truth labels [57], which in the context of supervised learning can be rephrased as the data scarcity problem.

As a result of the bottleneck, straightforward deep learning approaches are not always feasible for implementing a solution to a particular application. A typical solution to this bottleneck by practitioners is typically the reuse of a model derived from a related dataset [45], we can consider this an example of transfer learning. Another more recent approach originally presented by Fei-Fei et al. [14] and Fink [16] has introduced the concept of few-shot learning which proposes the idea of learning from a limited number of samples. Inspired by the human visual system and its remarkable ability to quickly and effortlessly adapt and recognize novel visual concepts after seeing just one or a few samples [35]. When reviewing the literature on few-shot learning in remote sensing [57], we noticed that there is a bias toward using meta-learning methods, with unfair comparisons being made to the baseline transfer learning methods. We cannot say the same for other domains, as there has been a lot of motivation for the use of transfer learning-based approaches over the more complex meta-learning-based approaches. To establish the general applicability of the methods regarding classification tasks in remote sensing a fair comparison is in order. We believe the concept of few-shot learning can be leveraged to democratize the use of deep learning methods globally, potentially improving the time to train and deploy models in data(label)-scarce scenarios.

Traditional deep learning models are generally data-driven methods, requiring substantial amounts of time, expert knowledge, and interpretation capacity to collect and annotate the volume of data required for these models [18]. These approaches are only capable of classifying classes that the model has been trained on but not novel classes. That in combination with inter-class similarity and intra-class variability motivates the research we conduct on few-shot learning methods in the remote sensing domain. With few-shot learning being the concept, transfer learning and meta-learning are the approaches that can be utilized by practitioners to bring the concept to life. It is worth mentioning that both approaches have been applied to tasks in the remote sensing domain both in regular supervised learning and few-shot learning settings [78, 38, 77, 5, 52]. However, to our knowledge, no fair comparison of transfer learning and meta-learning methods has been applied to the remote sensing domain, specifically remote sensing scene classification.

This thesis presents a systematic comparison that evaluates the performance of transfer learning and meta-learning in a few-shot learning setting, specific to remote sensing with the underlying research questions formulated as. How do transfer learning and meta-learning algorithms compare in terms of performance in a few-shot learning setting in the domain of remote sensing imagery? And which approach is most suitable for various real-world scenarios?. . Taking into consideration various real-world scenarios that practitioners are likely to face. This research does not seek to establish new model architectures for remote sensing but rather establish which method(s) are most suitable for few-shot learning in a remote sensing setting. We aim to contribute the following:

- By performing extensive experiments to evaluate the performance of the various approachesnamely, Baseline, Baseline++, Model-Agnostic Meta-Learning, Prototypical Networks, Matching Networks, and Relation Networks we aim to determine which methods are appropriate given the circumstances of the data and available resources.
- By conducting experiments to assess model performance in a standard few-shot setting and under more challenging conditions, including varying network depths, cross-domain data usage, and class imbalance settings, using the following diverse set of datasets, UC-Merced, NWPU-RESISC45, WHU-RS19, PatternNet, and Aerial Image Dataset.

We hypothesis that the transfer learning methods will be able to build a diverse feature set that can better generalize to new unseen tasks when compared to the complex meta-learning algorithms.

The remainder of this thesis is structured as follows. Section 2 presents background information relevant to the reader followed by the related works in Section 3 3 that discuss what has been done in various domains as well as the drawbacks in the research specific to few-shot learning in remote sensing image classification. Sections 4 and 5 present the methodology and experimental setup where we motivate the various experiments and how to make a fair comparison between the performance of the methods. Section 6 contains the results of the experiments followed by Section 7 where we discuss the concluding remarks and proposals for future works.

2 Background

In this section, we present general background information as well as several definitions relevant to the remainder of this thesis.

2.1 Remote Sensing

Remote sensing refers to the acquisition of information from a distance, which in the context of this thesis specifically refers to remote sensing imagery. Remote sensing imagery can be acquired through the means of satellites, aircraft, and/or drones. The data acquired can be utilized for data-driven decision-making as it provides perspective and a wealth of information about the Earth's surface.

2.2 Deep Learning

Classical machine learning methods benefit from domain-specific, hand-crafted features acting as proxies for dependencies in space and time. Deep learning methods on the other hand are capable of automatically extracting features, without the exhaustive feature selection and creation process. Helping to make deep learning methods the go-to approach for numerous tasks when given sufficiently large models, datasets, and/or labelled training examples [21]. There are many deep learning variants, however, we will remain focused on supervised learning methods. Referring to the use of labelled datasets, as opposed to the unsupervised and semi-supervised settings.

More specifically, supervised classification deals with a datasets $\mathcal{D} = \{\mathcal{D}_{train}, \mathcal{D}_{test}\}$, with the training set consisting of labeled pairs, $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^{N}$, $y \in \{C_1, .., C_N\}$ with N the number of training samples and C_{train} the number of categories in $\mathcal{D}_{train}$. The aim is to learn a function $\hat{y} = f_\theta(x)$ based on $\mathcal{D}_{train}$ to predict the labels $\hat{y} = \{C_1, .., C_N\}$ of the test set $\mathcal{D}_{test} = \{(x_k)\}_{k=1}^{K}$ with K the number of samples.

Convolutional Neural Networks: in deep learning, convolutional neural networks are a class of artificial neural networks, commonly applied to visual data. The innovation of convolutional neural networks was inspired by biological processes, with the connectivity between neurons resembling how the visual cortex of animals is structured. When compared with other image classification algorithms, convolutional neural networks use little manual pre-processing making the automated feature extraction a huge advantage over classical machine learning methods [21].

Convolutional neural networks are relevant in the domain of remote sensing due to the complexity of applications such as classification, object detection, super-resolution, change detection, and more. Aside from the complexity of the tasks themselves, there is also additional complexity when dealing with hyper/multi-spectral data and the integration of remote sensing data with other sources of information. Convolutional neural network's ability to process complex, high-dimensional data makes them ideal for the above-mentioned tasks.

Since deeper neural networks are more difficult to train, we use standard convolutional neural network architectures for the smaller model backbone and a residual learning framework for the deeper networks [25].

Domain Shift/Adaptation: When dealing with deep learning, the data used for training and testing share common characteristics and are drawn from the same distribution. Generally delivering impressive results, however, having training and testing data drawn from the same distribution is not always feasible in real-world applications. In many scenarios having access to complete data is not feasible, which is where domain shift/adaptation from here on out referred to as domain shift becomes relevant. In deep learning, domain shift occurs when the data used for training, validation, and testing are drawn from varying distributions. Domain shift can affect the performance of models on the test data. Common in the application of deep learning models, being able to measure a model's ability to be trained on one or multiple source domains to perform tasks different from the target domain is relevant for model selection and deployment. While domain shift/adaptation is not unique to deep learning models, it has become a popular research topic in the past years, especially in deep learning.

Class Imbalance: Class imbalance occurs whens the distribution of classes in a dataset is uneven, which is a significant challenge in training and testing machine learning models. It is widely recognize that the imbalance in class distribution can degrade the performance of deep learning models. This issue is particularly pronounced when one or more classes have significantly more instances than others. If not handled appropriately, the loss functions used during training of the models may become skewed, introducing biases that hinder the model's ability to generalize well to underrepresented, or minority classes [3]. In real-world environments, the distribution of categories follow an inherently long-tailed distribution, where a few classes occur many times, while many classes are scarcely represented [56]. There is no one-size-fits-all solution to address class imbalance; the optimal approach depends on the data and the specific task at hand.

Several approaches have been developed to cope with class imbalance, including data-level and method-level approaches. Data-level methods operate on training data, making changes to allow standard training algorithms to work. Algorithm-level methods, keep the training data unchanged and make relevant adjustments during the training or to the inference algorithms [24]. Methods that combine both of the above-mentioned approaches also exist [3]. Popular data-level approaches include random oversampling, while popular algorithm-level approaches include regularization or minimization of loss functions [33].

Random oversampling is a technique used for balancing datasets, the minority class is over-sampled by randomly duplicating samples of the minority class [33]. The goal is to augment the number of minority class samples to ensure that the model receives a balanced representation of all classes during training. Random under-sampling also exists, however, the focus is on reducing the number of majority classes to balance the class during training.

Re-balancing losses is another approach used to tackle class imbalance in deep learning. It functions by modifying the loss function used to train a model, assigning higher weights to the minority class samples. Examples of re-balancing loss functions include weighted loss [3, 37], focal loss [39], and class balancing loss [10]. This is not an exhaustive list of re-balancing loss functions however, it does cover the functions that are commonly used.

The data-level and algorithm-level approaches to tackling class imbalance have been described in a binary classification setting. However, these approaches can also be applied in a multi-class classification setting. When considering multi-class classification problems, it is important to note that class imbalance can occur in many ways. It may be the case that one class is under-represented or over-represented, it can also be the case that every class has varying data available.

2.3 Few-Shot Learning

Few-shot learning introduces various terms that will be new to readers, hence this section clarifies those that are relevant. We also provide some background details on transfer learning and meta-learning, two approaches that can be utilized in the few-shot learning paradigm. In general, few-shot learning seeks to enable recognition of new classes with only a few examples, a concept inspired by biological processes [19]. Few-shot learning aims to alleviate several issues faced by regular supervised learning methods, including the effort required to gather data, computational costs, and the time spent in training models [62]. Beyond the above-mentioned, models that generalize from a few examples would be able to generalize to data that is impossible or difficult to acquire due to privacy, safety, or rarity. Few-shot learning offers a solution here as it allows models to learn from a small number of labelled examples and generalize to new classes.

Given a dataset $\mathcal{D}_{train}$ consisting of image-label pairs sampled from a distribution $\mathcal{P}_{train}$, the objective is to learn a function f, which itself functions as a few-shot learner. The learner f uses samples from the same dataset but different instances, denoted $\mathcal{D}_{few}$, drawn from a new distribution $\mathcal{P}_{few}$ where $\mathcal{P}_{few} \neq \mathcal{P}_{train}$. The output of f is a classification function g, which is then used to classify samples from the original distribution $\mathcal{D}_{train}$. The few-shot setting arises when $\mathcal{D}_{few}$ contains only a small number of examples, often as little as one sample per class [9].

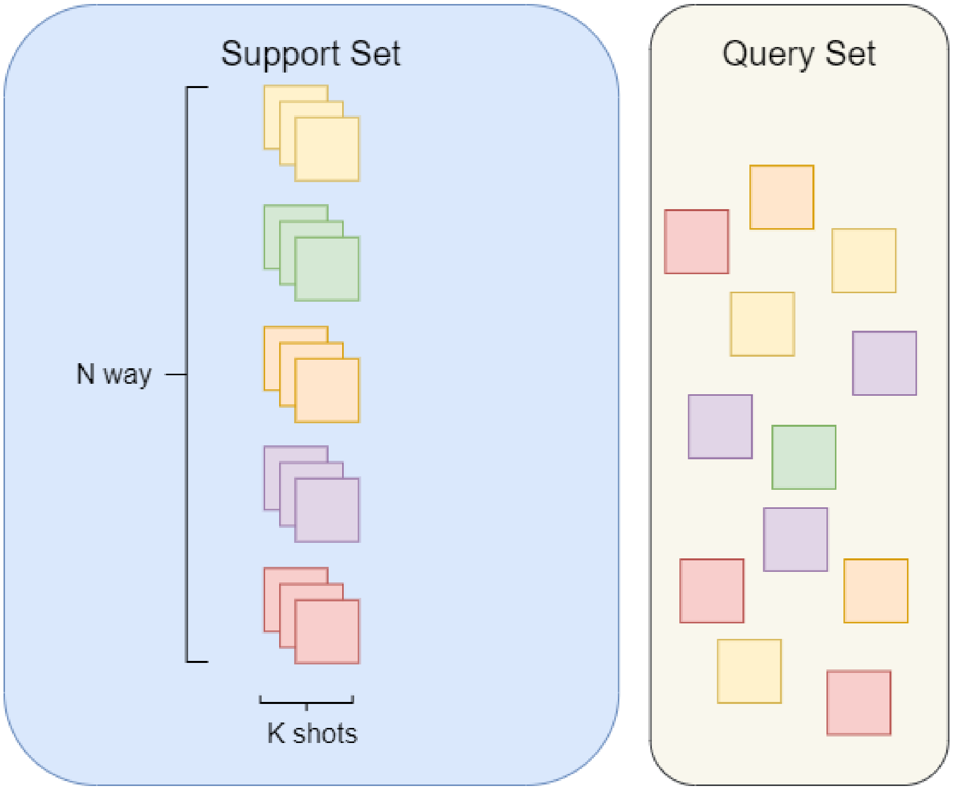

Figure 1: An example of a task in a few-shot learning setup which involves a support set and a query set, where n represents the number of classes and k refers to the number of instances per class. The query set can contain any number of class instances per class and is used to validate model performance during both meta-training and meta-testing. Graphic inspired by Ravi and Larochelle [55].

n-way-k-shot learning is a schema typically used in few-shot learning to describe the format of the few-shot learning problem at hand. n-way stands for the number of novel classes a pre-trained model needs to generalize over, while k-shot denotes the number of labeled samples available per class in n. The values for k usually range from one to five, and may also include zero. The problem then changes from a supervised to an unsupervised learning problem. It is important to note that in few-shot learning, the sets of classes encountered during training and testing are disjoint. In few-shot learning, datasets are split into what are known as tasks. The tasks are used to structure the data for different stages of the learning process. The splits are categorized into meta-training, meta-validation, and meta-testing sets with each of them containing multiple tasks as shown in Figure 2.

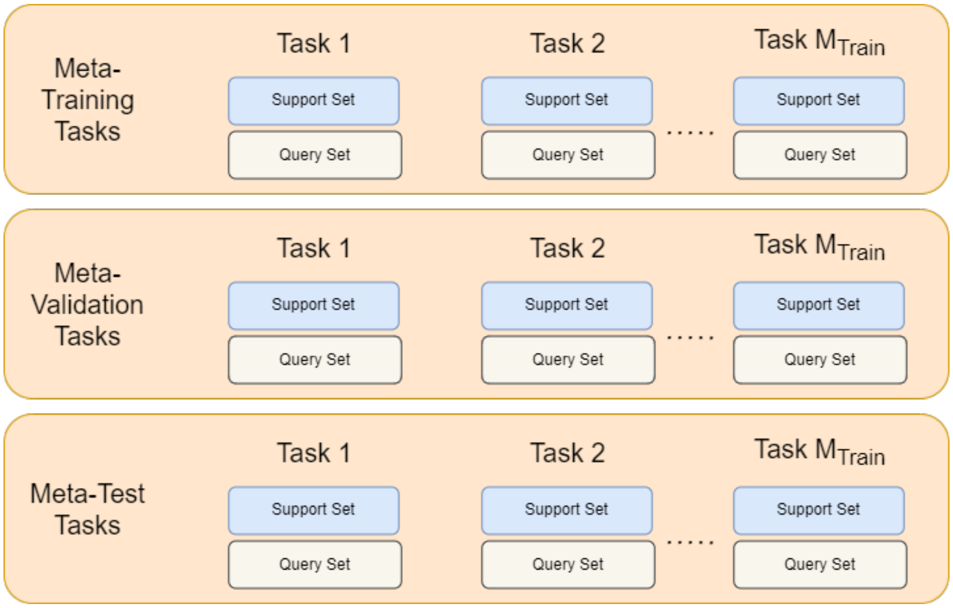

Figure 2: An example of a few-shot learning setup. During meta-training various tasks $(1, ..., M$, with M referring to the total number of tasks exposed to the model, typically 600) as illustrated in Figure 1 are used to train a model with a disjoint set of tasks sampled from the dataset used for validation. During meta-testing, the test tasks are used for the performance evaluation of the model. Graphic inspired by Ravi and Larochelle [55].

Transfer Learning Transfer learning is a concept where knowledge gained from solving one problem is used to address a similar or related problem [49]. Transfer learning takes advantage of large feature maps, without the need to train a network entirely from scratch for a new task. Transfer learning is a broader concept and is not typically considered as an approach to few-shot learning however, transfer learning can be applied in data-scarce settings (few-shot learning). The main distinction between transfer learning and few-shot learning lies in the training methodologies. Transfer learning, often referred to as pre-training and fine-tuning, involves pre-training a model on a large dataset (upstream) and then fine-tuning it for a related downstream task. This reduces the number of samples required for training on the target dataset [46].

More specifically, given a large and abundant set of base classes, X_b (with b referring to the classes used for training) and a small volume of novel classes X_n (with n referring to the classes used for meta-testing), a few-shot learner aims to train a model to recognize novel classes, which were not seen during training, using only a few examples [4]. few-shot learning focuses on rapidly training models that can adapt to new tasks with only a few examples, typically through meta-learning algorithms, the approach to training is quite different in the scenes that the training setup mimics the setting used for testing. In other works the models aren't exposed to the entire dataset for training but rather samples of the data [64].

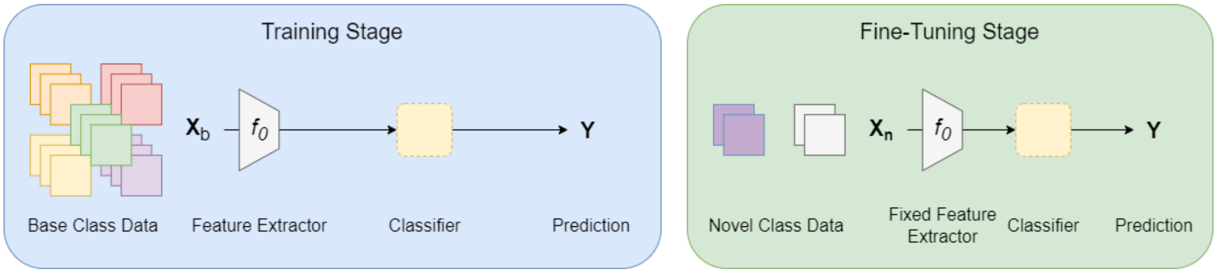

Figure 3: Example of the transfer-learning process. Graphic inspired by Chen et al. [4].

Transfer learning, as illustrated in Figure 3, involves two stages: pre-training and fine-tuning. During pre-training, both the feature extractor and classifier are trained from scratch by minimizing the loss with examples from the base classes $X_i \in X_b$. In the fine-tuning stage, the model is adjusted to identify new classes. Here, the pre-trained parameters θ of the feature extractor remain unchanged, and a new classifier is trained by minimizing the loss using a small set of labeled examples from the support set X_n.

During the training process of the transfer learning methods, the data is divided into batches, following a general non-episodic training process. The batches consist of a fixed number of input-output pairs, which the training algorithm iterates over. These iterations over the training data by the algorithm are known as epochs.

Meta-Learning goes beyond the concept of just learning tasks but it is designed to "learn how to learn" new tasks. The primary objective of meta-learning is to train a model on a diverse range of tasks so that it can effectively tackle new tasks with only a minimal amount of training samples [17]. Meta-learning algorithms are organized into two stages: meta-training and meta-testing. In the context of transfer learning, meta-training corresponds to the pre-training phase, during which the model is trained on the complete dataset. Meta-testing, on the other hand, corresponds to the fine-tuning phase in few-shot learning, which is specifically defined as the meta-testing tasks in Figure 2

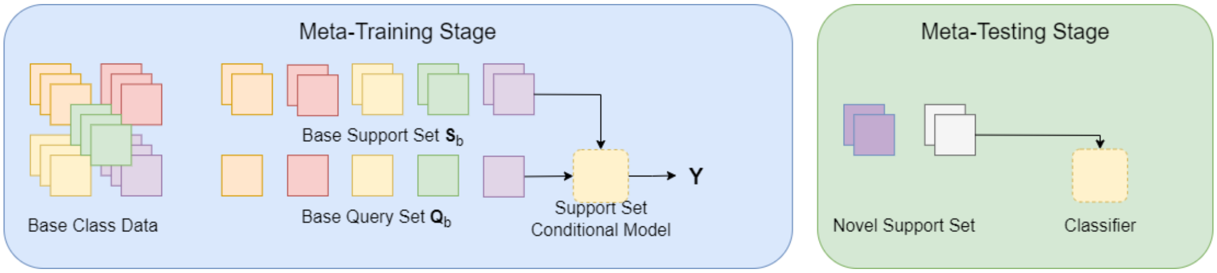

Figure 4: Example of the meta-learning process. Graphic inspired by Chen et al. [4].

As shown in Figure 4, the meta-training stage aims to develop a model M that leverages the base support set S_b. During this phase, the model focuses on minimizing the n-way prediction loss for samples from the base query set Q_b. This process allows the meta-learner to acquire the ability to learn from limited labeled data by training across various tasks. In the meta-testing stage, the model is presented with novel classes X_n, which constitute the new support set S_n. The model M is then fine-tuned to predict the new classes using the updated support set S_n [4].

Meta-learning methods can broadly be grouped into metric-based [70], optimization-based [17], and several other approaches. Approaches that will not be covered in this research include hallucination-based methods [79] and probabilistic-based approaches. Hallucination-based methods augment the data by generating addition example and probabilistic-based methods attempt to model uncertainty.

Meta-learning methods use episodic training (tasks) unlike the transfer learning methods that are trained in epochs. Episodic training mimics the evaluation conditions of the meta-learning models, meaning a small random sample of the training data (support set) is used to train the model with another disjoint subset of the data (query set) used for model testing [70, 17, 55]. Epochs refer to a pass through the entire dataset during model training. In the context of this thesis, models are trained using epochs where for both the transfer learning and meta-learning algorithms, however, the epochs for the meta-learning algorithm consist of many tasks instead of the entire training dataset. The meta-testing format stays the same for both the transfer learning and meta-learning algorithms (i.e., model evaluation consists of the meta-testing format as defined in Figure 2).

3 Related Work

Few-shot learning has received extensive interest over the past few years, especially for computer vision tasks, specifically scene classification [40, 61], image segmentation [72], and object detection [73, 31]. While the majority of previous works focus on natural scene imagery datasets such as CUB [71] and ImageNet [12]. There has been an uptick in the volume of research focusing on few-shot learning in the remote sensing domain [57, 18, 78, 38, 47]. Nearly all of the remote sensing domain-specific research explores new meta-learning approaches as opposed to evaluating how current methods can be utilized to achieve similar results. Likely motivated by various factors, such as the cost of data acquisition and labelling as well as the diversity of the Earth's surface [57]. The authors of Ma et al. [43] have shown that in recent years the main focus of deep learning for remote sensing applications revolved around classification tasks; namely land-use land-cover classification, object detection, and scene classification. They also show that other applications are worth mentioning namely, fusion, segmentation, change detection, and registration. The broader deep learning paradigm has been applied to all aspects of the remote sensing domain, while the exploration of few-shot learning in remote sensing is still in the early stages.

Various works have shown that convolutional neural network-based methods are capable of obtaining impressive results when presented with several data samples for fine-tuning and training a network from scratch [8]. Penatti et al. [51] evaluate the performance of convolutional neural networks training on natural scene images for the classification of remote sensing imagery, showing the generalization power of convolutional neural networks even when there is a domain shift between the training and test data.

Researchers have realized the potential of deep learning for remote sensing and have developed an open-source Python library TorchGeo [63] used to integrate geospatial data into the PyTorch [50] framework. Salinas et al. [58] have also integrated remote sensing data into the Auto-Keras deep learning framework showing how existing systems can be leveraged for remote sensing.

3.1 Transfer Learning

In this thesis , we take inspiration from research that has been done in both the natural image scene as well as remote sensing scene. Several recent works have suggested that transfer learning can outperform more sophisticated meta-learning methods for tasks in few-shot learning, through the re-use of high-quality, pre-trained feature extractors [32].

Chen et al. [4] have shown that transfer learning, namely methods baseline and baseline++ achieves performance comparable to the top performing meta-learning algorithms including ProtoNet, MAML, RelationNet and MatchingNet on several benchmark few-shot learning natural scene image datasets including CUB [71] and Mini-ImageNet [70]. They specifically evaluate the performance of few-shot learners when faced with a domain shift between the base and novel classes. Chowdhury et al. [9] examine the use of a library of convolutional neural networks as a basis for developing high-quality few-shot learners, bench-marked across different image datasets. Not only do they show that using a learner built on a high-quality feature extractor can produce performance on par with a state-of-the-art meta-learner but they go as far as to show that an ensemble of transfer learners can produce results significantly better than any other few-shot learning method. Another comparative study of few-shot learning methods by Dumoulin et al. [13] has shown that large-scale transfer learning methods using the standard pre-training and fine-tuning procedure as mentioned in Section 2outperform competing methods (i.e. meta-learners). Tian et al. [66] also show that baseline transfer learning methods are capable of outperforming state-of-the-art meta-learning methods, with further improvements being achievable through self-distillation. These works show the power of transfer learning, while their focus is on natural scene imagery it is believed that the generalization power of transfer learning methods can extend to the remote sensing domain.

Rußwurm et al. [57] use transfer learning as a baseline for both few-shot classification and segmentation tasks in remote sensing imagery, making use of 7-layer convolutional neural networks. Neumann et al. [46] explore transfer learning on remote sensing datasets, showing that when limited training samples are present, performance enhancements can be observed when using in-domain data compared to training models from scratch or fine-tuning only on ImageNet.

3.2 Meta-Learning

Meta-learning being very topical in deep learning has received a lot of attention in recent years, namely methods such as Model-Agnostic Meta-Learning, Prototypical Networks, Matching Networks, and Relational Networks have been created to tackle various tasks in a few-shot setting. While most research is done concerning image classification tasks in natural scene imagery, researchers have also applied meta-learning in a remote sensing setting [77, 78, 38, 5]. it is important to mention that applications of meta-learning extend beyond image classification tasks, however, our research focuses on remote sensing tasks for which classification models have the potential to have a large impact. Meta-learning methods have been applied to various remote sensing tasks motivated by the varying distribution of data in the remote sensing domain. As mentioned, the majority of research done concerning meta-learning and image classification revolves around natural scene imagery. Many of the papers that use transfer learning as a baseline use meta-learning methods as state-of-the-art approaches [4, 9, 66].

Rußwurm et al. [57] use transfer learning as a baseline for two remote sensing tasks on data collected through the Sentinel 1 and 2 satellites. They evaluate the performance of Model-Agnostic Meta-Learning on classification and segmentation tasks using globally and regionally distributed datasets. Rußwurm et al. [57] were motivated to use Model-Agnostic Meta-Learning as the model is design to adapt and learn new tasks from minimal amounts of data, by training the model in such a way that when new tasks are shown it can update its model parameters in only a few gradient steps. Their results indicate that model-agnostic meta-learning outperforms pre-training and fine-tuning on various datasets when the source and target domains differ. Arguments can be made that they didn't use a strong enough feature extractor to fairly evaluate the performance of the transfer learner and that their choice of backbone network could have been the reason for the under performance of the baseline method. Zeng and Geng [78] and Li et al. [38] both developed a novel meta-learning approach to few-shot learning for remote sensing-specific tasks on several relevant remote sensing datasets. Including NWPU-RESISC45 [7], UC-Merced [76], and WHU-RS19 [74]. They also present dataset splits for those datasets which are adopted in this research.

3.3 Domain Shift/Adaptation

There have been several cross-domain studies done in both the standard supervised learning setting as well as in the few-shot learning setting. Guo et al. [23] developed a cross-domain few-shot learning benchmark that consists of data acquired from diverse methods. Including natural scene, satellite, dermatology, and radiology images. They conduct comprehensive cross-domain experiments on the benchmark dataset to evaluate both cutting-edge meta-learning methods and transfer learning approaches. Their findings reveal that, within the cross-domain context, transfer learning methods exceed the performance of current state-of-the-art meta-learning techniques. Ullah et al. [68] have curated a meta album of approximately 40 datasets intended for few-shot learning, meta-learning, and continual learning. van den Nieuwenhuijzen et al. [69] perform extensive experimentation to evaluate how well pre-trained models can be fine-tuned to tasks from other datasets. More specifically, they group and evaluate the curated datasets of meta-album into three distinct groups, same-source, same-domain, and cross-domain to determine where performance improvements can be derived from.

Neumann et al. [46] investigate the suitable characteristics of a diverse set of remote-sensing datasets for representation learning. While they don't specifically investigate these datasets in a few-shot learning setting, it shows the practical application of evaluating the cross-domain setting as it is a circumstance faced by remote sensing practitioners. The conclusion drawn from their paper indicated that using in-domain knowledge can assist in improving model performance as opposed to using out-of-domain knowledge or training from scratch. Motiian et al. [45] contributed early to cross-domain research in a few-shot learning setting developing an approach to unsupervised and supervised domain adaptation. They specifically focus on the case where there are few labelled samples available for training. Hsu et al. [26] introduce a novel method for performing transfer learning across domains. While their method is novel and complex focusing on unsupervised learning, the main contribution is once again showing the power of transferring knowledge between domains. Setlur et al. [60] explore meta-learning methods in a few-shot learning setting, by sampling tasks from in-distribution and out-of-distribution. Their results showcase challenges such as what models to use and how to make a fair comparison when dealing with an out-of-distribution setting. As previously mentioned, Penatti et al. [51] evaluate the generalization capability of convolutional networks trained on natural scene images and fine-tuned on aerial and remote sensing image classification datasets. Their results indicate that transfer learning models can generalize data from different domains.

3.4 Class Imbalance

The related works on class imbalance are available in the standard supervised learning setting [3, 30] as well as in the few-shot learning setting [48, 59, 41, 67, 36, 6, 22]. With class imbalance being a common issue when dealing with classification tasks, especially when considering multi-class classification tasks we see quite a bit of attention that has been given to class imbalance in the context of few-shot learning. There have been several methods created to cope with the challenges it brings, including data-level and algorithm-level approaches. Many approaches stem from research in the standard supervised learning setting. Johnson and Khoshgoftaar [30] surveyed fifteen studies between 2015 and 2018 showing that traditional machine learning techniques for handling class imbalance can have a positive impact on deep learning models, specifically computer vision models. Given the extensive research on the impact of class imbalance on classification tasks, we will focus the remainder of the related works on studies that address the few-shot learning setting. Polak et al. [53] also provide a literature review however theirs focuses on works addressing two issues in low-shot learning. Namely, the obstacles of class imbalance, or utilizing low-shot learning techniques or frameworks to combat class imbalance.

Triantafillou et al. [67] introduced a meta-dataset introducing a realistic class imbalance setting in the few-shot learning domain. The dataset provides a varying number of classes for each task and training set, allowing researchers to test the robustness of models across several realistic settings. The work done by Lee et al. [36] and Chen et al. [6] make use of imbalanced tasks in their experiments but they don't make any reference to how the imbalance in the tasks or dataset affects model performance. They only try to replicate a more realistic few-shot learning setting. Guan et al. [22] proposed a strategy to train a model when dealing with class-imbalanced remote sensing imagery. Their approach involved two phases: a random episodic training phase and an all-classes fine-tuning stage. Overall, their approach is capable of improving the recognition accuracy of the minority classes with only a slight decrease in accuracy of the majority classes on the Aerial Image Dataset [75] and NWPU-RESISC45 [7] datasets. Ochal et al. [48] provide a survey paper that examines class imbalance in a few-shot learning setting. Exploring varying distributions of class imbalance in both task and dataset level imbalance on state-of-the-art few-shot image classification techniques and evaluating how countermeasures to class imbalance affect model performance. Their findings indicate that transfer-learning algorithms typically outperform state-of-the-art meta-learning methods.

With the majority of related works suggesting that complex meta-learning methods outperform transfer learning methods, this thesis aims to provide a comparative analysis of few-shot learning methods under a controlled experiment setting to provide a clearer understanding of the strengths and weaknesses of transfer learning and meta-learning methods. The majority of related works comparing complex meta-learning to few-shot learning methods don't typically provide a fairground for comparison with the transfer learning methods. For example, they typically evaluate model performance on simple backbone networks (Conv4) while it is fair to assume that deeper networks provide transfer learning methods with a wealth of information for further inference. The explored methods include transfer learning methods baseline and baseline++ and meta-learning methods MAML, ProtoNet, MatchingNet, and RelationNet. These methods were chosen as they are well established and have been shown to produce competitive performance in the tasks of natural scene classification in a few-shot setting. More specifically, we want to understand how the challenges of remote sensing imagery can be tackled at scale.

4 Methodology

This research addresses the few-shot recognition problem in an image classification setting. A broad set of established few-shot learning methods are adopted into the experimental framework for evaluation. The models chose represent a diverse group of few-shot learning methods from transfer learning to meta-learning methods. More specifically, the meta-learners were chosen as they have proven to be effective in a few-shot setting This includes both traditional transfer learning approaches, which are often used as baselines, and advanced meta-learning methods that represent state-of-the-art techniques in the field. We aim to compare the performance of the various few-shot learners to have a holistic overview of their application in the remote sensing domain. We do this by studying not only the effectiveness of the methods under different circumstances but also by examining the data itself. Having an understanding of the data will allow us to define which methods are suitable in different scenarios. We do this by comparing models in different experimental setups across 5 diverse yet well-established remote sensing datasets.

4.1 Transfer Learning Based Approaches

Both Baseline [4] and Baseline++ [4, 20, 54] follow a standard transfer learning procedure, which involves pre-training followed by fine-tuning. Initially, both methods train a feature extractor and a classifier using the base dataset. In the fine-tuning stage, the feature extractor remains unchanged, while a newly initialized classifier is trained on the novel dataset [20]. The output of the new classifier is tailored to the number of classes in the new task (n-way), with fine-tuning conducted using the support set. Baseline trains a feature extractor and classifier based on a linear layer and a softmax layer while Baseline++ trains a feature extractor and classifier based on a Cosine distance layer and softmax layer [20].

4.2 Meta-Learning Based Approaches

Similar to [4, 48], this research considers an optimization-based method Model-Agnostic Meta-Learning (MAML) [17] and metric-based methods including Prototypical Networks (ProtoNet) [61], Relation Networks (RelationNet) [65], and Matching Networks (MatchingNet) [70]. As mentioned in Section 2 meta-learners consist of a meta-training phase and a meta-testing phase, with the former consisting of several base classes n, with a base support set and a base query set. During the meta-testing phase, the novel classes are used as the support set and the model is then adapted to classify novel classes using the new support set.

MAML is an optimization-based algorithm that seeks to achieve a good initialization of its parameters instead of learning an update function or learning rule. It learns a set of weights over training tasks that can be adapted to new unseen tasks with a few gradient steps. The adjustment process employs a standard gradient descent algorithm to minimize the loss on the support set [17]. We use the first-order approximation, which has been shown to provide similar performance while using far less memory. ProtoNet is a metric-based meta-learning algorithm that functions similarly to a nearest neighbour classifier, by mapping the input into a feature space and classifying instances based on some distance function. ProtoNet first estimates the class mean vectors (prototypes) over the support set, and then it measures the distance between a query set and the prototypes [61]. The query samples are subsequently classified by measuring their Euclidean similarity to the prototypes. Similar to ProtoNet, RelationNet follows the same process but instead of calculating the Euclidean distance of an input sample to a class mean a CNN-based relation module is used to assign the class [65]. A relation module is used to compare the embeddings of the support set with the embeddings of the query set. A relation score between 0 and 1 represents the similarity between the query and support samples[65]. MatchingNet differs in its process from both ProtoNet and RelationNet in that the query features are compared to each support feature instead of the class means. Using the context embeddings with an LSTM, MatchingNet performs k-nearest neighbours classification with Cosine similarity as a distance metric to classify the query set [70].

5 Experiments

This section presents the various datasets, experiments formulated, and metrics used to tackle our research questions.

5.1 Datasets

This section lists several remote sensing benchmark datasets used to evaluate the performance of the various few-shot learning methods. Table 1 shows an overview of the datasets and their characteristics. The dataset splits are also provided in Table 2, and the splits for NWPU-RESISC45, WHU-RS19, and UC-Merced are taken from previous works that perform few-shot learning on remote sensing datasets [18, 38].

Dataset	**Characteristics**					**Task**
	Categories	**Images**	**Resolution (m)**	**Size**	**Bands**	
UC-Merced	21	2100	0.3	256x256	RGB (3)	Scene classification
NWPU-RESISC45	45	31500	0.2-30	256x256	RGB (3)	Scene classification
WHU-RS19	19	1005	0.5	600x600	RGB (3)	Scene classification
PatternNet	38	30400	0.06-5	256x256	RGB (3)	Scene classification
AID	30	10000	0.5-8	600x600	RGB (3)	Scene classification

Table 1: An overview of all datasets featured in this thesis with their corresponding characteristics.

Dataset	Training	Validation	Testing
UC-Merced	Agricultural; Baseball diamond; Buildings; Chaparral; Dense residential; Freeway; Harbor; Medium residential; Overpass; Parking lot	Airplane; Forest; Intersection; Runway; Storage tanks	Beach; Golf course; Mobile home park; River; sparse residential; Tennis court
NWPU-RESISC45	Airplane; Baseball diamond; Beach; Bridge; Chaparral; Church; Cloud; Desert; Freeway; Golf course; Harbor; Island; Lake; Meadow; Mobile home park; Mountain; Palace; Railway; Rectangular farmland; Roundabout; Sea ice; Ship; Sparse residential; Stadium; Wetland	Commercial area; Industrial area; Overpass; Railway station; Runway; Snowberg; Storage tank; Tennis court; Terrace; Thermal power station	Airport; Basketball court; Circular farmland; Dense residential; Forest; Ground track field; Intersection; Medium residential; Parking lot; River
WHU-RS19	Airport; Bridge; Desert; Football field; Industrial; Mountain; Parking; Port; Residential	Beach; Farmland; Forest; Park; Railway station	Commercial; Meadow; Pond; River; Viaduct
PatternNet	Airplane; Baseball field; Beach; Bridge; Cemetery; Chaparral; Christmas tree farm; Freeway; Golf course; Harbor; Mobile home park; Ferry terminal; Football field Parking space; Railway; Coastal mansion; Closed road; Runway marking; Sparse residential	Overpass; Crosswalk; Oil gas field; Runway; Storage tank; Shipping yard; Solar panel; Swimming pool Tennis court; Transformer station; Oil well	Basketball court; Dense residential; Forest; Intersection; Parking lot; Nursing home; Wastewater treatment place; River
AID	Agricultural; Beach; River; Forest; Harbor; Baseball diamond; Chaparral; Intersection; Parking lot; Medium residential	Storage tanks; Dense residential; Airplane; Overpass; Golf course; Buildings	Freeway; Sparse residential; Tennis court; Runway; Mobile home park

Table 2: Description of the remote sensing dataset train, validation, and test splits.

UC-Merced [76] dataset consists of a selection of manually extracted images from urban areas around the United States, a country typically well represented in open-source datasets. The dataset consists of 21 classes with 100 samples per class, making it a balanced dataset. The dataset consists of a variety of land-use patterns with overlapping classes for example the types of residential classes, which only vary based on the density of the structure, making the classification task more challenging [75]. UC-Merced has also previously been used to evaluate the performance of few-shot learning methods [1].

NWPU-RESISC45 (RESISC45 [7] is another large-scale remote sensing imagery dataset collected through Google Earth Engine. RESISC45 contains the largest number of categories, providing rich image variations and high within-class diversity and between-class similarity similar to the Aerial Image Dataset. The dataset consists of 45 classes with 700 images per class, making it a balanced dataset. RESISC45 has also been used to evaluate the performance of few-shot learning methods [1].

WHU-RS19 [74] is another remote sensing imagery dataset collected through Google Earth Engine. Providing high-resolution satellite imagery, with image samples of the same class collected from different regions, resolutions, scales, and orientations. The dataset consists of 19 classes with 50-70 samples per class, making it an imbalanced dataset.

PatternNet [80] is a large-scale, high-resolution, multi-source dataset with images collected through Google Earth Engine and via the Google Maps API. Similarly to the UC-Merced dataset, the dataset consists of images representing the United States. The dataset consists of 38 classes with 800 samples per class, making it a balanced dataset.

Aerial Image Dataset (AID) [75] is a large-scale aerial image dataset obtained through Google Earth Engine imagery, making it a multi-source dataset, adding a layer of complexity for classification when compared to using single source datasets. Unlike UC-Merced, AID consists of sample images selected from different countries and regions around the world. Not only are the locations different but so are the seasons and imaging conditions, increasing the intra-class diversity. The dataset consists of 30 classes with 200-400 samples per class of differing resolutions, making it an inherently imbalanced dataset. AID has also previously been used to evaluate the performance of few-shot learning methods [1].

5.2 Performance Evaluation

We use accuracy as the primary metric to evaluate the performance of the various methods. Accuracy is the most commonly used metric for evaluating classification tasks in a few-shot setting. The average accuracy is used to evaluate the performance of the methods across several n and k typically between {1, 2, ..., 10} n and {1, ..., 5} k, indicating how the performance changes due to the availability of data. For image classification tasks, 5-shot accuracy scores tend to outperform the 1-shot accuracy scores. Indicating, that data scarcity is a large bottleneck for achieving good performance [27] which in the remote sensing setting would be the lack of ground truth labels. Typically a higher n means a more difficult task and with a low k the few-shot learning task becomes more difficult because less supporting information is available to draw an inference.

Each experiment is run 5 times with different random seeds to determine the statistical significance of the differences in performance of the various methods. Each model is trained for 200 epochs, with 200 tasks used for validation and 600 tasks used for evaluation. The evaluation tasks consist of 15 randomly sampled samples of each class. A 95% confidence interval is combined with an evaluation of confusion matrices to evaluate model performance.

While running statistical significance tests using both the Wilcoxon Signed-Rank Test and the Friedman Test, we were unable to identify any statistical significance between the models across the different datasets. As a result, no identifier is used on the results tables and the analysis of the results is based solely on the average model performance and the variance, not any statistical difference.

5.3 Experimental Setup

The experiments being carried out for this systematic comparison are based on previous works evaluating the performance of few-shot learning methods as well as new experiments, namely class imbalance. The code for reproducing the experiments is available on GitHub [1]

Standard setting

In the standard setting, the performance of the few-shot learning methods is measured across varying n as mentioned above. Helping to verify the correctness of the implementations of the various methods, and providing a general overview of their performance in the remote sensing domain. As proposed by the authors of [4] and [61], a four-layered convolutional backbone (Conv-4) along with an input layer of 84 x 84 is used to conduct experiments in a common few-shot learning setting.

Research question: How do the various few-shot learning methods perform when supported by a simple feature extractor, namely a Conv4 backbone network?

Network depth

To overcome some of the potential issues a shallow backbone network has on the performance of the various few-shot learning methods, experiments with network depth are executed to provide an overview of how the few-shot learning methods adapt given more powerful and flexible backbones. The backbone networks are based on convolutional networks as well as residual networks.

[1] https://github.com/hennything/rsfsl/tree/main

The experiments are extended by using a six-layer (Conv-6) convolutional network, as well as three residual network backbones of ten-layers (ResNet-10), eighteen-layers (ResNet-18), and thirty-four-layers (ResNet-34). The models utilizing a Conv backbone have an 84 x 84 input layer, while the models that have a ResNet backbone use a 224 x 224 input layer.

Research question: What is the trade-off between network depth and model performance?

Cross-Domain

Cross-domain evaluation is relevant as it facilitates the understanding of how domain shift affects few-shot learning methods in a remote sensing setting. Having models that can be trained on a single dataset yet perform tasks on another alleviates various bottlenecks. Another consideration for the practicality of evaluating few-shot learning methods in a cross-domain setting is the fact that data from a general class may be easily collected but more specific hard-to-come-by classes may be difficult to collect data for. Cross-domain evaluation of few-shot learning methods has only been explored in the domain of natural scene imagery.

Research question: How is model performance affected when the feature extractor is trained on one dataset and evaluated on another?

Class imbalance

Class imbalance has been known to harm supervised learning tasks. Methods have been established to cope with class imbalance, however, the exploration of the effects of class imbalance in few-shot learning is limited. The standard training procedure for few-shot learning models typically does not account for real-world scenarios where classes appear with varying frequencies. Class imbalance can occur on the task level, dataset level, and as a combination of the two [48].

Research questions: How resilient are the various few-shot learning approaches to class imbalance? Are some models naturally resistant to class imbalance and/or do general approaches to resolving class imbalance have an impact on the performance?

5.4 Implementation Details

During the training stage the models are trained for a total of 200 epochs, the best performing epoch if saved during training by monitoring the accuracy on the validation set by sampling 200 tasks.

During the fine-tuning and meta-testing stages, the results are averaged over 600 tasks with different random initialization. For the transfer learning methods, the entire support set is used to train a new classifier for 100 iterations with a batch size of 4 as done by Chen et al. [4].

The few-shot learning methods are trained from scratch with general data augmentation being applied to the input images. The augmentation includes random cropping, left-right flip, and colour jitter to both the training and meta-training stages. During the class imbalance experiments, we do not apply data augmentation during the initial training phase, providing an overview of how the standard and augmented random over-sampling (ROS/ROS+) strategy impacts the model performance under various settings. Some small implementation details are adjusted individually for some of the few-shot learning methods described by [4, 48]. All methods use an Adam optimizer with a learning rate of 10^{-3}.

We provide a brief overview of the model-specific implementation details:

Baseline: is the typical way of performing transfer learning. the network is initially pre-trained on a large dataset and then fine-tuned on a smaller, domain-specific dataset [49]. The pre-trained backbone network is complemented by a single linear layer, which is subsequently replaced and retrained with a new linear layer that corresponds to the number of classes in the specific task during the n-way fine-tuning process [48].

Baseline++: Baseline++ is different in comparison to Baseline in that a cosine distance metric is used between the input features and the weights for each class on the last layer [4].

MAML: a meta-learning algorithm that aims to initialize parameters for a base model such that after applying a few gradient steps using the training set, the model can adapt and achieve generalization performance on the validation set [2]. The idea is to make the base model as general as possible so that it can be used to find solutions for many tasks quickly. As for the method-specific implementation details, we use the first-order approximation in the gradient for MAML. This is done for memory efficiency and has been shown to produce identical performance [27].

ProtoNet: a meta-learning algorithm aims to train a classifier by learning a metric space that can be used for classification by computing distances to prototype representations of each class [61]. The classification is based on a Euclidean distance metric, mapping query samples into their respective prototypes [48].

MatchingNet: a meta-learning algorithm that uses an attention mechanism to classify new examples by comparing them to a labelled support set. Using a combination of bi-directional Long Short-Term Memory and a Cosine similarity measure to map the support and query sets into an embedding space [70].

RelationNet: a meta-learning algorithm that learns to learn a distance metric to compare the images within episodes. The relation network classifies new images by computing relation scores between query images and the examples of the new classes without updating the network [65].

To get an overview of how the models perform in a cross-domain setting, we train them using the same datasets used in the standard and network-depth settings. The models utilize a Conv-4 backbone network for cross-domain experiments.

Standard few-shot learning assumes equal class distributions, however, as previously mentioned this is rarely the case in real-world scenarios [47, 22, 44]. Datasets can be combined or swapped with domain-specific datasets that are tailored for particular applications, especially when dealing with class distributions that are imbalanced. In a few-shot setting, the imbalance can appear on the task level, dataset level, or as a combination of the two. Figure 5 depicts a balanced setting, while Figures 7 and 6 depict task level and dataset level imbalance. Figure 8 depicts a combination of task level and dataset level imbalance.

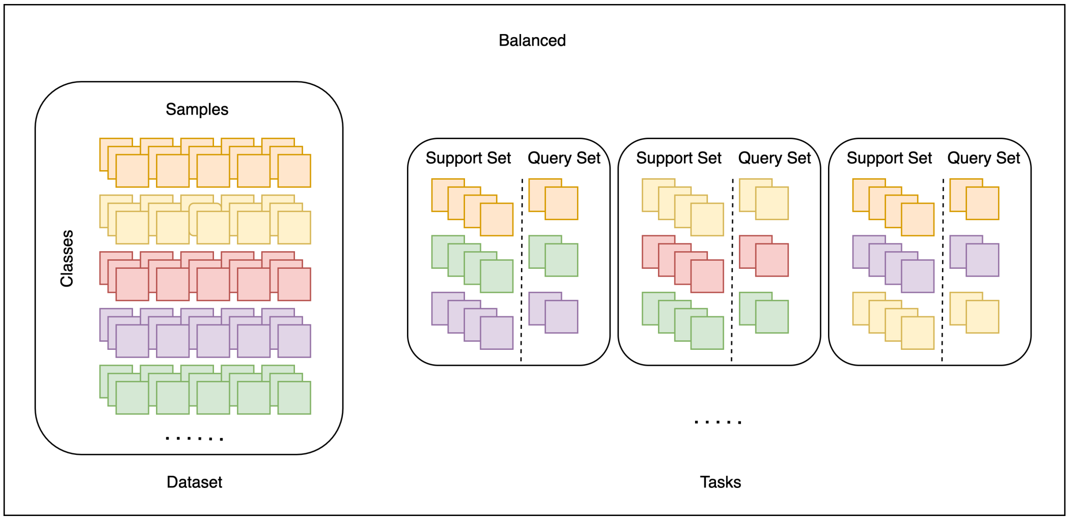

Figure 5: Balanced dataset/tasks. Graphic inspired by Ochal et al. [48].

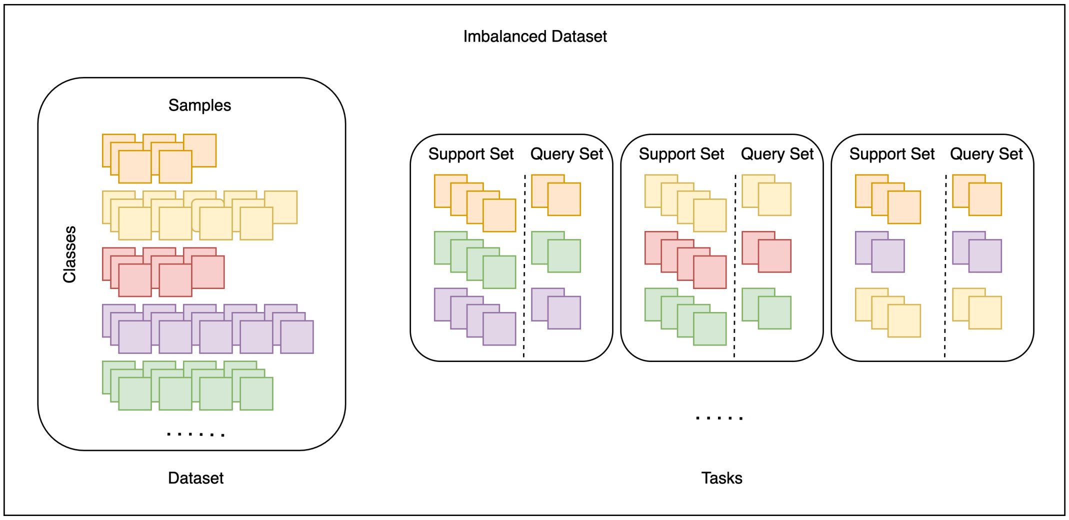

Figure 6: Imbalanced dataset. Graphic inspired by Ochal et al. [48].

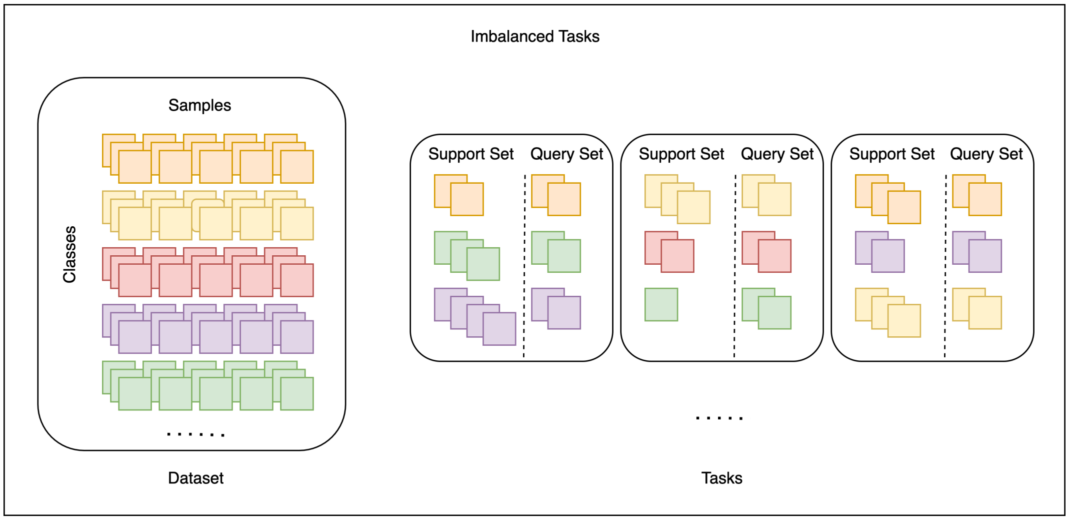

Figure 7: Imbalanced tasks. Graphic inspired by Ochal et al. [48].

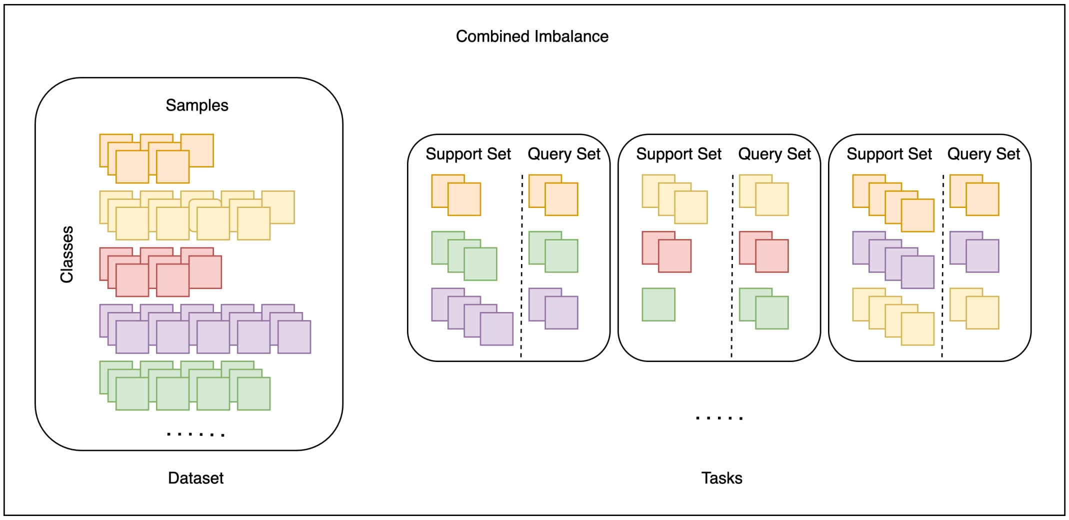

Figure 8: Imbalanced dataset and tasks. Graphic inspired by Ochal et al. [48].

Class imbalance is prompted into the datasets to get an overview of how class imbalance affects the performance of the various few-shot learning methods. Class imbalance is prompted into the datasets using linear, step, and random distributions. Linear imbalance is characterized by a single parameter: the ratio between the maximum and minimum number of examples across all classes. The number of examples in the intermediate classes is linearly interpolated to ensure that the difference between consecutive class pairs remains constant [3]. Step imbalance is defined by two parameters: the fraction of minority classes and the ratio of examples in minority classes to those in majority classes [3]. Random imbalance involves determining the number of samples per class through random uniform selection.

The distributions are defined using a tuple as provided by [48]. The tuple is defined as $(K^*_{min}, K^*_{max}, N^*, M^*)$, with K_{min} referring to the minimum number of samples per class, K_{max} referring to the maximum number of samples per class, with N^* as the number of classes and M^* as an additional parameter used for step imbalance. Equation 1 presents the formula used to determine the class distribution for linear imbalance. The class distribution for step imbalance is presented in Equation 2. Equation 3 presents the formula used to determine the class distribution for random imbalance.

$$K^*_i = round(K^*_{min} - c + (i-1) * (K^*_{max} + 2 * c + K^*_{min})/(N^* - 1)) \tag{1}$$

$$K_i^* = \begin{cases} K_{min}^*, & if\, i \leq M \\ K_{max}^*, & otherwise \end{cases} \tag{2}$$

$$Unif(K_{min}^*, K_{max}^*) \tag{3}$$

The class imbalance techniques and strategies that are used to tackle the issue of data imbalance include random over-sampling without data augmentation (ROS) and random over-sampling with data augmentation (ROS+). We focus solely on using data-level class imbalance strategies as they have been shown to be more effective than method-level approaches [11, 15]. The models utilize a Conv-4 backbone network for the class imbalance experiments with the backbone being trained on non-augmented data (unlike in the other experiment settings). This is done to provide an overview of how much impact the various settings have and how the class imbalance coping strategies can affect the performance.

Verification of implementation: The few-shot learning methods are implemented in PyTorch, adapting the implementations of [4, 48], as well as other openly available implementations. The code is suited to run on both CPU and GPU machines. The methods are first verified on Mini-ImageNet [70] using the standard 5-way-5-shot task setup. The results of the verification are presented in Table 10 in Appendix A. The results of the experiments on Mini-ImageNet are compared against the results of other papers using the same settings.

Limitations: While running the network depth experiments we ran into memory issues during the training of MAML. The initial assumption was that we were using second-order MAML, however, we double-checked the implementation and ensured that we are indeed using first-order MAML. As a result the network depth experiments in Section 6 only depict the performance of MAML using shallow networks (up to Conv6). Another caveat we have to mention is that the network depth experiments use different architectures as the network depth increases (Conv and ResNet), we have chosen to use different architectures because of the exploring/vanishing gradient problem faced by convolution neural network architectures as the depth of the network increases. Residual Networks have proven to be more resistant to the exploding/vanishing gradient problem. It is also important to mention that we do not perform hyperparameter optimization on the models during training, the hyperparameters used may not represent the best possible configuration. We are aware that hyperparameter optimization could potentially increase the performance of various models, however, we chose not to include it in our work as we already perform extensive experiments.

6 Results

In this section, we systematically present and evaluate the results of the various experiments. The implementation of the various few-shot learning methods as previously mentioned are first verified on Mini-ImageNet to ensure that the methods are properly implemented.

6.1 Standard Setting

In this section we attempt to answer research question 1, how do the various few-shot learning methods perform when supported by a simple feature extractor? The results in this section are significant for the remainder of the thesis as it provides a Baseline for fair comparison under different experiment settings.

As expected model accuracy improves when increasing the number of training samples (1 vs 5 shots) allowing models to learn more effectively and improve performance. The results in Table 3 suggest that MatchingNet and RelationNet models are most effective in this low-shot setting for datasets RESISC45, UC-Merced, and WHU-RS19. The transfer learning methods outperform the meta-learners on PatternNet and AID. The varying performance of the models on the different datasets makes it challenging to determine which method is suitable for which type of dataset. While RESISC45 and PatternNet are both larger datasets with varying image resolutions the model's performance under the low-shot conditions varies greatly. If we look at the aggregate performance of the models across the various datasets, we see Baseline++ has the highest overall accuracy with 63% followed by MatchingNet and RelationNet with 60% each respectively. The results in Table 4 suggest that ProtoNet, Baseline, and Baseline++ are most effective in a more sample-rich setting. The Baseline transfer learning method also under-performs on RESISC45 as it did in the 1-shot setting. The performance of RelationNet seems to lag behind the other models in a more complex sample-rich setting. The standard setting considers a small backbone network, the smaller backbone network affects the Baseline models by not allowing them to gather enough information about the various classes. When ranking the performance of the models across all datasets, we see that both transfer learning methods and ProtoNet have performance on par with one another at 80% accuracy. As mentioned in Section 5, we ran statistical significance tests using both the Wilcoxon Signed-Rank Test and the Friedman Test, and we were unable to identify any statistical significance between the models. Hence, the best performing models under the different settings are indicated based purely on model accuracy.

Model	Dataset				
	RESISC45	UC-Merced	WHU-RS19	PatternNet	AID
Baseline	$0.485_{\pm0.0072}$	$0.448_{\pm0.0065}$	$0.665_{\pm0.0064}$	$0.747_{\pm0.0069}$	$0.566_{\pm0.0077}$
Baseline++	$0.561_{\pm0.0078}$	$0.470_{\pm0.0068}$	$0.705_{\pm0.0063}$	$\mathbf{0.789_{\pm0.0069}}$	$\mathbf{0.624_{\pm0.0082}}$
MAML	$0.532_{\pm0.0088}$	$0.512_{\pm0.0080}$	$0.687_{\pm0.0067}$	$0.683_{\pm0.0079}$	$0.485_{\pm0.0073}$
MatchingNet	$0.587_{\pm0.0082}$	$\mathbf{0.523_{\pm0.0072}}$	$\mathbf{0.733_{\pm0.0063}}$	$0.665_{\pm0.0064}$	$0.480_{\pm0.0078}$
ProtoNet	$0.515_{\pm0.0084}$	$0.504_{\pm0.0073}$	$0.705_{\pm0.0060}$	$0.634_{\pm0.0079}$	$0.440_{\pm0.0074}$
RelationNet	$\mathbf{0.613_{\pm0.0086}}$	$0.522_{\pm0.0078}$	$0.715_{\pm0.0064}$	$0.670_{\pm0.0071}$	$0.461_{\pm0.0077}$

Table 3: 5-way-1-shot with a simple backbone network (Conv4)

Model	Dataset				
	RESISC45	UC-Merced	WHU-RS19	PatternNet	AID
Baseline	$0.699_{\pm0.0063}$	$0.679_{\pm0.0053}$	$\mathbf{0.897_{\pm0.0030}}$	$0.940_{\pm0.0033}$	$\mathbf{0.790_{\pm0.0031}}$
Baseline++	$0.733_{\pm0.0062}$	$0.655_{\pm0.0054}$	$0.848_{\pm0.0034}$	$\mathbf{0.942_{\pm0.0037}}$	$\mathbf{0.790_{\pm0.0030}}$
MAML	$0.731_{\pm0.0068}$	$0.673_{\pm0.0061}$	$0.829_{\pm0.0041}$	$0.876_{\pm0.0052}$	$0.665_{\pm0.0072}$
MatchingNet	$0.718_{\pm0.0066}$	$0.674_{\pm0.0056}$	$0.850_{\pm0.0035}$	$0.939_{\pm0.0032}$	$0.719_{\pm0.0067}$
ProtoNet	$\mathbf{0.748_{\pm0.0063}}$	$\mathbf{0.689_{\pm0.0055}}$	$0.869_{\pm0.0032}$	$0.938_{\pm0.0032}$	$0.755_{\pm0.0031}$
RelationNet	$\mathbf{0.748_{\pm0.0063}}$	$0.639_{\pm0.0051}$	$0.814_{\pm0.0034}$	$0.865_{\pm0.0054}$	$0.655_{\pm0.0034}$

Table 4: 5-way-5-shot with a simple backbone network (Conv4)

6.2 Network Depth

In this section we attempt to answer research question 2, what is the trade-off between network depth and model performance? This section should indicate how network depth impacts model performance.

One would expect the performance of all models to improve when increase the complexity or size of the feature extractor. As seen in the Figures 9-13 the first thing we notice is that the Baseline transfer learning method is consistently one of the top performing models, typically showing consistent improvement as the backbone complexity increases. RelationNet is consistently the worst-performing model across all datasets. This contradicts the claims made by several previous works [61, 70, 46, 55]that favour the use of complex meta-learning methods in a few-shot learning setting. The results are presented with the caveat that the architectures of the shallower backbone networks (ConvNet) differ to those of the deeper networks (ResNet), however residual networks are used for the deeper networks to overcome the vanishing/exploding gradient problem [25].

The size and variety of the data must be concerned when evaluating the performance of the models on the datasets. UC-Merced and WHU-RS19 contain only 2,100 images with 21 classes and 1,005 images for 19 classes respectively as opposed to the RESISC45 dataset which contains 31,500 images for 45 classes and PatternNet with 30,400 images for 38 classes. AID provides a nice middle ground for comparison with a total of 10,000 images across 30 classes. We don't see a significant increase in model performance for UC-Merced and WHU-RS19 however, we do see that model performance improves as the backbone complexity increases.

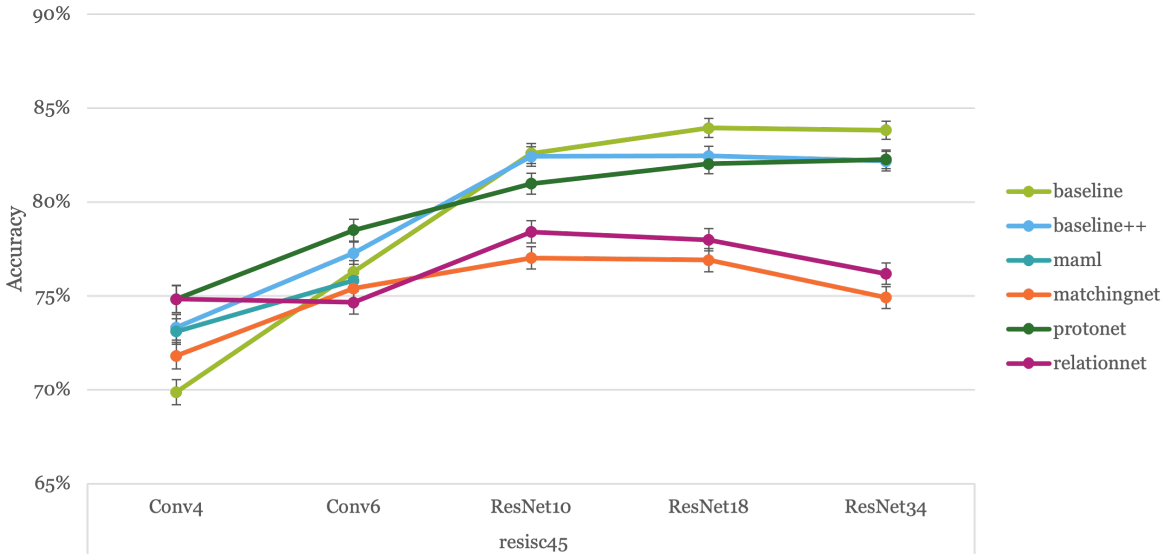

Figure 9: Results on increasing the network depth on the RESISC45 dataset.

When focusing on the results of the RESISC45 dataset in Figure 9, we see that Baseline, Baseline++ and ProtoNet are performing on par with one another. At the same time, the other few-shot learning models lag in terms of model accuracy, even with increased network depth. With the RESISC45 dataset, which consists of 31,500 images distributed equally across 45 classes, we see that model performance typically increases as the complexity of the backbone network increases. RESISC45 has a high level of intra-class variability, due to the seasonal changes and variations in the spatial resolution of the images. It also has inter-class similarity, making it a more challenging dataset because many of the classes are visually similar. Baseline, Baseline++, and ProtoNet adjust to the difficulty of the task as the network depth increases showing that they are capable of generalizing and differentiating the subtle differences of varying classes, while other few-shot learners struggle to do the same.

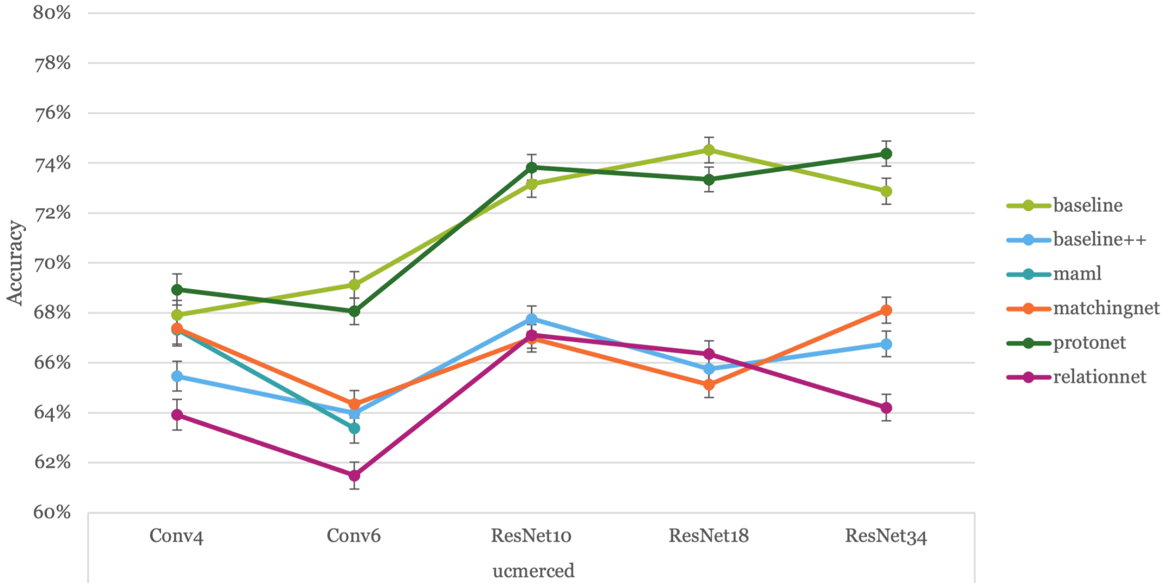

Figure 10: Results on increasing the network depth on the UC-Merced dataset.

In Figure 10 we see a clear gap between the performance of Baseline and ProtoNet and the other models, not only do they show improvement as the network depth increases but the gap between Baseline and ProtoNet and other models continues to grow as the network depth increases. Unlike RESISC45, UC-Merced has a lower intra-class variability due to the source of the dataset, with variation for the same classes arising due to seasonal conditions. However, it also has the challenges of inter-class similarity with several classes being visually similar (e.g., sparse residential, medium residential, and dense residential). With UC-Merced being a much smaller dataset in comparison to RESISC45, the models may also face the challenge of having less information to train on.

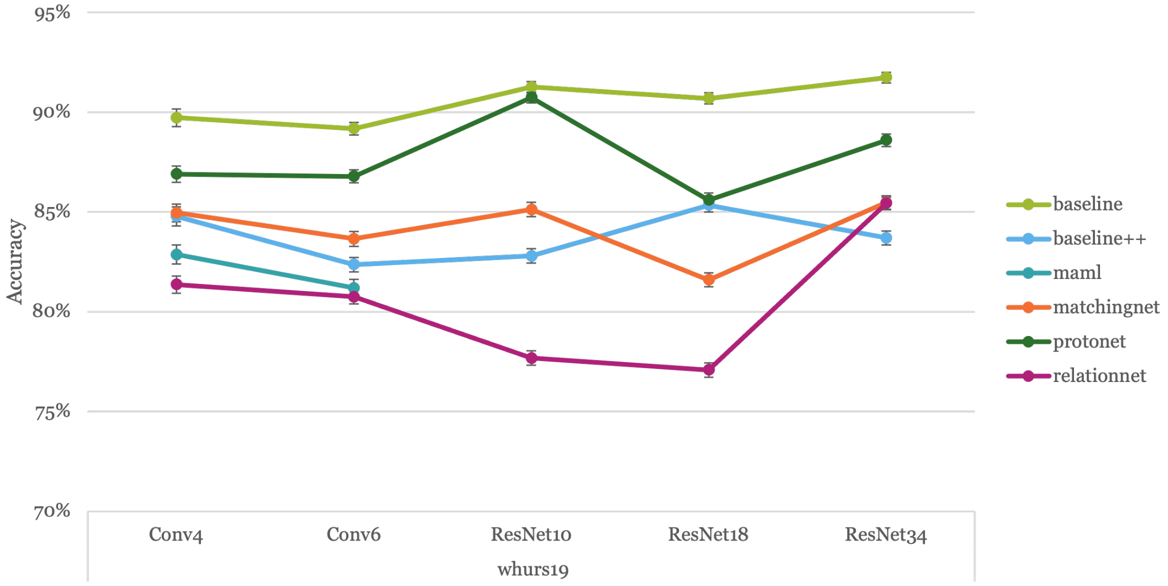

Figure 11: Results on increasing the network depth on the WHU-RS19 dataset

In Figure 11 Baseline outperforms all other models at all levels of network complexity, however, ProtoNet is a close performer. WHU-RS19 has a low inter-class similarity, with all classes being distinct from one another, however, it is another small dataset similar to UC-Merced but with an imbalanced class distribution.

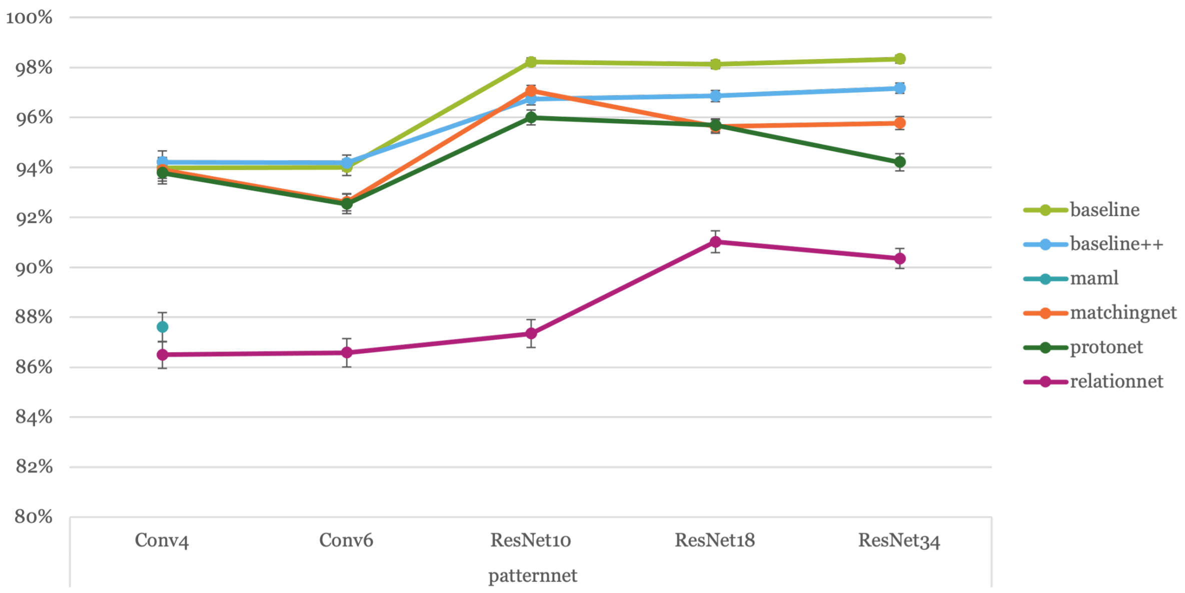

Figure 12: Results on increasing the network depth on the PatternNet dataset

Model performance on PatternNet (Figure 12) is quite different when compared to the other datasets. Instead of Baseline and ProtoNet being top performers we see several few-shot learners that are competitive. Baseline and Baseline++ are more stable as the network depth increases while MatchingNet and ProtoNet show more volatility. PatternNet is one of the larger datasets providing a more diverse class set which typically makes it a more challenging task for generalization, however with more moderate levels of inter/intra-class variability we see that the models are more stable and capable of generalizing quite well.

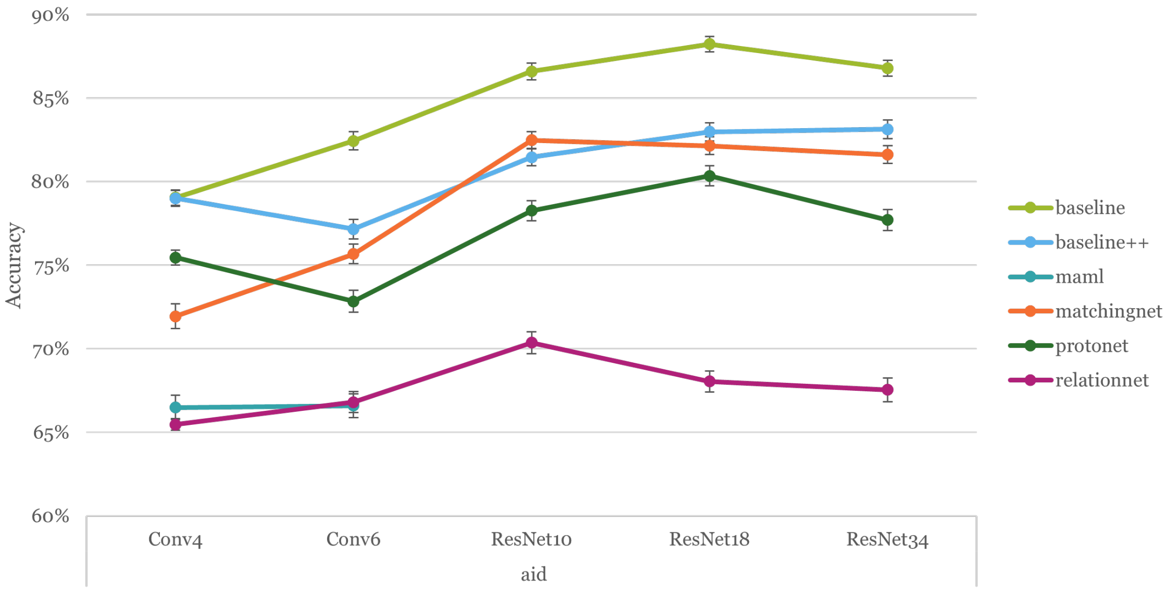

Figure 13: Results on increasing the network depth on the AID dataset

As with the RESISC45 dataset, AID also has a high inter/intra-class variability making it a difficult classification task. In Table 13, we see that the Baseline transfer learning method differentiates itself from the other few-shot learning models. This suggests that with added network complexity Baseline is better able to differentiate between the subtle differences between similar classes.

What we see in Figures 9-13 is that the Baseline transfer learning method is more stable as the network depth changes when compared to the meta-learning algorithms. As the network depth increases Baseline is still able to leverage the learned features from the datasets, the model's pre-trained layer thus provides a strong foundation, potentially reducing over-fitting. The meta-learning algorithms are less stable during training as the network depth increases, this may be due to their underlying design. When averaging the performance of the models across all datasets and network depths, Baseline and ProtoNet are top performers with 82% and 81% overall accuracy respectively. While we see that Baseline and ProtoNet have similar overall accuracy scores, it is important to note that the baseline transfer learning method typically performs worse than other models with shallower networks. The increased network complexity provide the transfer learning methods with a wealth of information for inference later on.

6.3 Cross Domain

In this section we attempt to answer research question 3, how is model performance affected when the feature extractor is trained on one dataset and evaluated on another? This section should indicate how well the models can transfer knowledge from one dataset to another. Model performance need to be compares to the model performance in Table 4 to understand how the training on one dataset impacts the model when evaluated on another, the results in this section indicate model performance using a standard Conv4 backbone network.

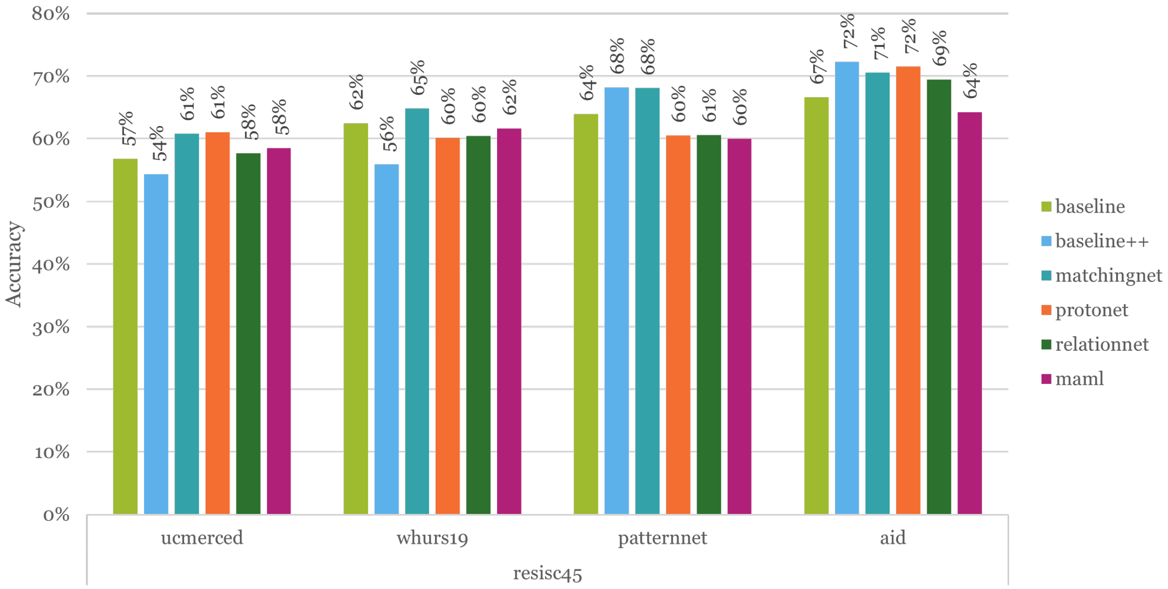

Figure 14: Results on training backbone networks on RESISC45 and testing on other datasets

In Figure 14 we see that training the models on RESISC45 and testing it on the other datasets doesn't provide any performance improvements. The performance of the models drastically decreases when compared to the results in Table 4, this can be due to several reasons stemming from the high intra-class variability of the dataset which may lead to poor generalization of the models to the variability in the image resolutions. When comparing the models across the different datasets, there is no clear indicator for which of the models perform well when trained on RESISC45 and tested on the other datasets. Meta-learning methods MatchingNet has the best overall performance with approximately 66% accuracy with the other models performing on par with one another at around 62%-63% (excluding MAML).

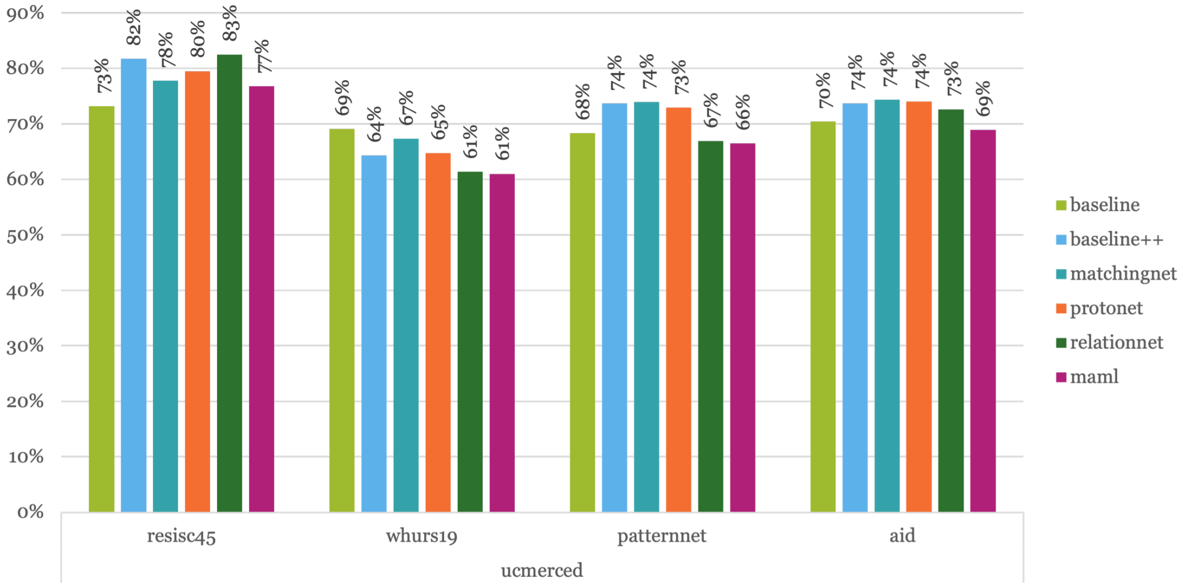

Figure 15: Results on training backbone networks on UC-Merced and testing on other datasets

When models are trained on UC-Merced (Figure 15), we see model performance increase for all models in the RESISC45 case and for all meta-learners in the case of AID excluding ProtoNet, while models tested on PatternNet and WHU-RS19 face a significant performance decrease. Baseline shows the least amount of improvement on the RESISC45 dataset with a performance improvement of approximately 3% and Baseline++ shows the largest improvement with approximately a 9% increase. Baseline++, ProtoNet, and MatchingNet have an overall performance on par with one another with approximately 73% overall accuracy.

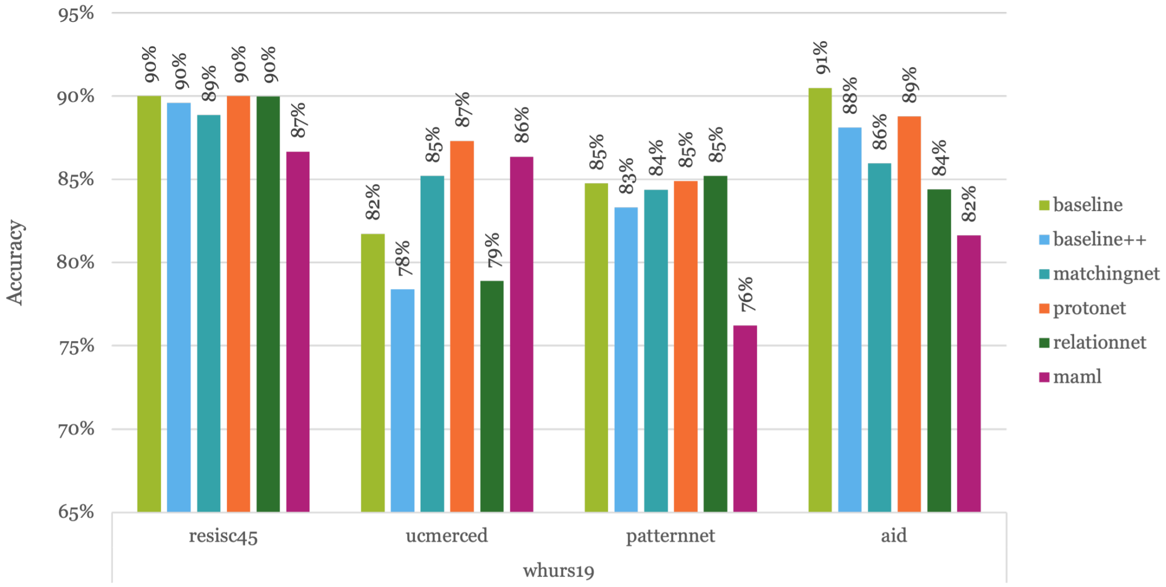

Figure 16: Results on training backbone networks on WHU-RS19 and testing on other datasets

When the model backbones are trained on WHU-RS19 we see that the performance of the models tends to increase model accuracy quite drastically across all datasets except when PatternNet is used for testing (Figure 16). Baseline and ProtoNet have the best overall accuracy scores across all datasets with an average of approximately 87% and 88% respectively.

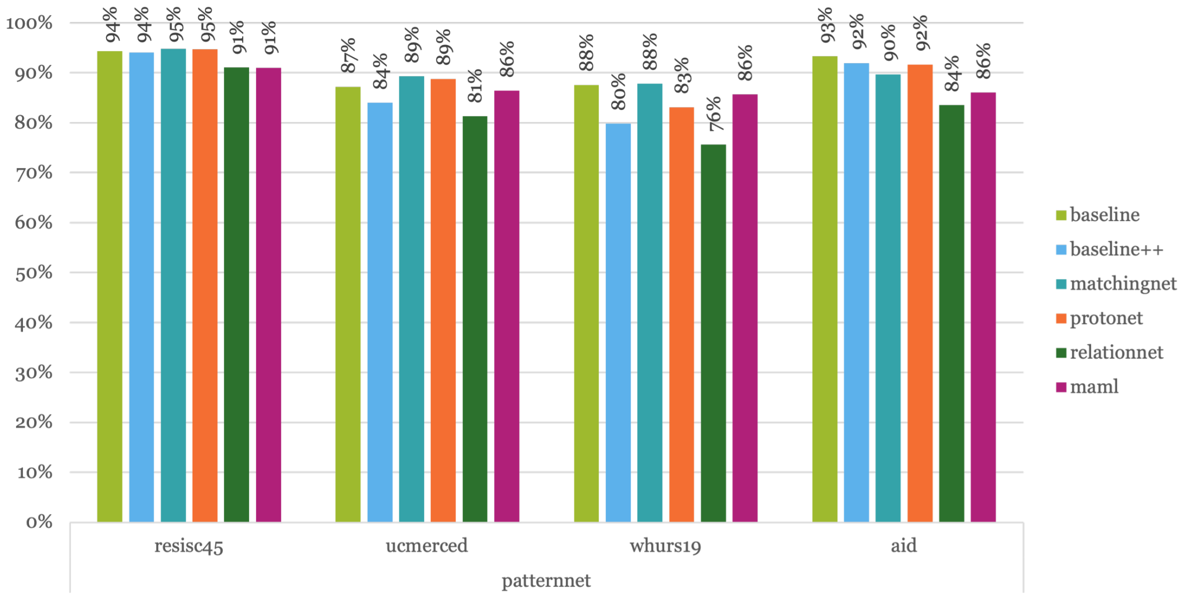

Figure 17: Results on training backbone networks on PatternNet and testing on other datasets

With PatternNet being one of the larger datasets we see in Figure 17 that the models benefit from the large and high-resolution dataset. With the high-resolution and diverse class representation, the models are provided with suitable information to generalize across domains. Baseline and MatchingNet have the best overall performance with approximately 91% accuracy followed closely by ProtoNet with 90%.

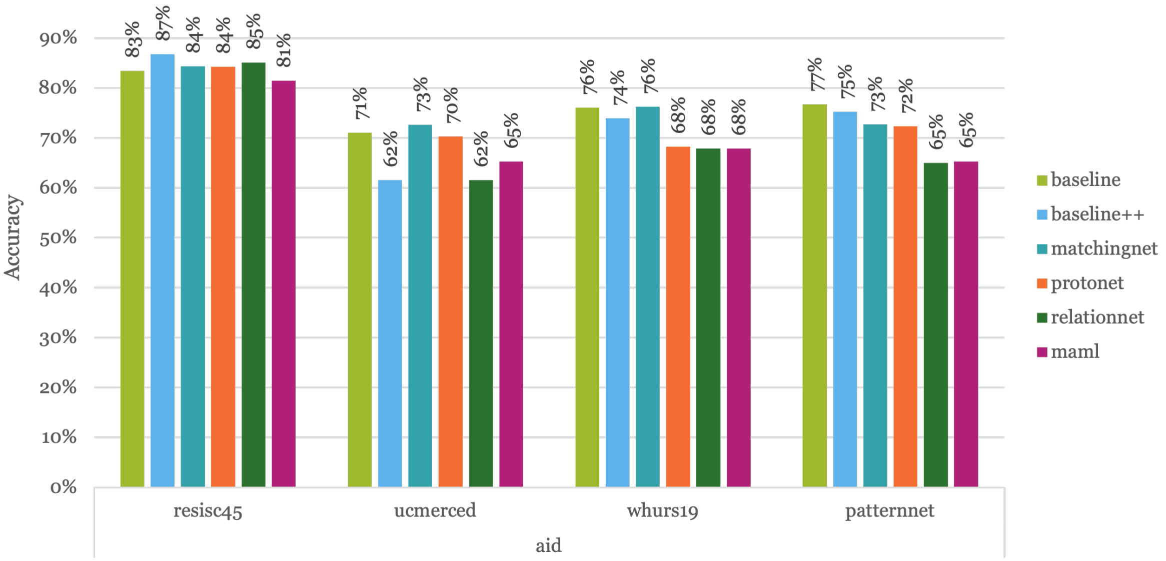

Figure 18: Results on training backbone networks on AID and testing on other datasets

In Figure 18 we see that the models trained on the AID dataset and tested on the RESISC45 dataset benefit the most while the models that are tested on WHU-RS19 and PatternNet have significant decreases in model performance. Once again we see Baseline and MatchingNet have an overall performance on par with one another with approximately 77% overall accuracy.

Based on the results of the cross-domain settings in Figured 14 - 18, it is difficult to make significant conclusions about which models are better able to generalize when training on one dataset and testing on another. When ranking the models based on their overall performance, we see that the two meta-learning algorithms MatchingNet and ProtoNet rank first and second respectively followed by the two transfer learning methods Baseline and Baseline++. RelationNet and MAML are tied for last with the worst overall performance. When using shallower backbone networks for model training and evaluation, meta-learning algorithms are the preferred methods to choose from.

6.4 Class Imbalance

In this section, we attempt to answer research questions 4 and 5. How resilient are the various few-shot learning approaches to class imbalance? Are some models naturally resistant to class imbalance and/or do general approaches to resolving class imbalance have an impact on their performance? With class imbalance occurring so prominently in nature, this section should provide an indication of which models are most resilient as well as how to overcome class imbalance in practice.

Tables 9 - 7 show the performance of the models across the different datasets under varying conditions, namely different shots and re-sampling strategies. The results indicate the performance of the models in a typical few-shot learning setting without ROS and ROS+ (random over sampling without/with augmentation), 4-6 shot Linear, 1-9 shot Step, and 1-9 shot Random settings. The base model is trained without data augmentation applied to the training data, hence the results differ to those listed in Table 4. As with the results in Section 6.1, we ran statistical significance tests using both the Wilcoxon Signed-Rank Test and the Friedman Test, and we were unable to identify any statistical significance between the models. Hence, the best performing models under the different settings are indicated based purely on model accuracy.

Model	Strategy	5-way-5-shot Train	4-6 shot Linear	1-9 shot Step	1-9 shot Random
Baseline	None	$0.712_{\pm 0.0053}$	$0.690_{\pm 0.0062}$	$0.653_{\pm 0.0039}$	$0.585_{\pm 0.0120}$
	ROS	-	$0.694_{\pm 0.0061}$	$0.659_{\pm 0.0044}$	$0.605_{\pm 0.0110}$
	ROS+	-	$0.708_{\pm 0.0058}$	$0.706_{\pm 0.0057}$	$0.676_{\pm 0.0075}$
Baseline++	None	$0.726_{\pm 0.0048}$	$0.700_{\pm 0.0069}$	$0.651_{\pm 0.0042}$	$0.589_{\pm 0.0130}$
	ROS	-	$\mathbf{0.722_{\pm 0.0062}}$	$0.692_{\pm 0.0062}$	$0.677_{\pm 0.0092}$
	ROS+	-	$0.724_{\pm 0.0065}$	$\mathbf{0.728_{\pm 0.0067}}$	$\mathbf{0.711_{\pm 0.0080}}$
MAML	None	$0.425_{\pm 0.0085}$	$0.425_{\pm 0.0083}$	$0.408_{\pm 0.0072}$	$0.396_{\pm 0.0076}$
	ROS	-	$0.425_{\pm 0.0084}$	$0.420_{\pm 0.0081}$	$0.406_{\pm 0.0086}$
	ROS+	-	$0.414_{\pm 0.0083}$	$0.416_{\pm 0.0090}$	$0.400_{\pm 0.0092}$
MatchingNet	None	$0.704_{\pm 0.0073}$	$0.698_{\pm 0.0072}$	$0.665_{\pm 0.0069}$	$0.659_{\pm 0.0096}$
	ROS	-	$0.702_{\pm 0.0072}$	$\mathbf{0.699_{\pm 0.0072}}$	$\mathbf{0.682_{\pm 0.0080}}$
	ROS+	-	$0.702_{\pm 0.0069}$	$0.699_{\pm 0.0071}$	$0.683_{\pm 0.0081}$
ProtoNet	None	$0.742_{\pm 0.0066}$	$\mathbf{0.732_{\pm 0.0063}}$	$0.672_{\pm 0.0048}$	$0.652_{\pm 0.0110}$
	ROS	-	$0.717_{\pm 0.0070}$	$0.672_{\pm 0.0048}$	$0.635_{\pm 0.0091}$
	ROS+	-	$\mathbf{0.734_{\pm 0.0064}}$	$0.721_{\pm 0.0063}$	$0.704_{\pm 0.0076}$
RelationNet	None	$0.732_{\pm 0.0067}$	$0.726_{\pm 0.0063}$	$\mathbf{0.675_{\pm 0.0056}}$	$\mathbf{0.668_{\pm 0.0095}}$
	ROS	-	$0.716_{\pm 0.0069}$	$0.675_{\pm 0.0056}$	$0.657_{\pm 0.0097}$
	ROS+	-	$0.723_{\pm 0.0067}$	$0.718_{\pm 0.0069}$	$0.696_{\pm 0.0078}$

Table 5: Results on task-level class imbalance on the RESISC45 dataset.

As expected model accuracy decreases as the complexity of the tasks increases however, we also see that ROS and ROS+ improve performance, especially in more complex scenarios. When comparing the performance of the models across different task-level imbalance settings in Table 5, we see that the meta-learning methods (excluding MAML) are more robust in dealing with imbalanced data when compared to the transfer learning methods. Once the over-sampling strategies are applied the discrepancy in performance becomes less pronounced, especially between Baseline++ and the meta-learners.

Model	Strategy	5-way-5-shot Train	4-6 shot Linear	1-9 shot Step	1-9 shot Random
Baseline	None	$0.687_{\pm0.0032}$	$0.683_{\pm0.0037}$	$0.637_{\pm0.0027}$	$0.587_{\pm0.0091}$
	ROS	-	$0.677_{\pm0.0036}$	$0.661_{\pm0.0032}$	$0.607_{\pm0.0080}$
	ROS+	-	$0.683_{\pm0.0037}$	$0.690_{\pm0.0034}$	$0.661_{\pm0.0054}$
Baseline++	None	$0.635_{\pm0.0033}$	$0.613_{\pm0.0032}$	$0.585_{\pm0.0033}$	$0.526_{\pm0.0075}$
	ROS	-	$0.628_{\pm0.0033}$	$0.618_{\pm0.0033}$	$0.593_{\pm0.0049}$
	ROS+	-	$0.633_{\pm0.0033}$	$0.644_{\pm0.0036}$	$0.619_{\pm0.0044}$
MAML	None	$0.542_{\pm0.0057}$	$0.532_{\pm0.0056}$	$0.506_{\pm0.0054}$	$0.487_{\pm0.0073}$
	ROS	-	$0.532_{\pm0.0059}$	$0.526_{\pm0.0054}$	$0.506_{\pm0.0060}$
	ROS+	-	$0.520_{\pm0.0060}$	$0.531_{\pm0.0069}$	$0.496_{\pm0.0072}$
MatchingNet	None	$0.670_{\pm0.0058}$	$0.661_{\pm0.0037}$	$0.641_{\pm0.0035}$	$0.624_{\pm0.0062}$
	ROS	-	$0.665_{\pm0.0036}$	$\mathbf{0.673_{\pm0.0036}}$	$\mathbf{0.647_{\pm0.0051}}$
	ROS+	-	$0.663_{\pm0.0036}$	$0.677_{\pm0.0039}$	$0.649_{\pm0.0051}$
ProtoNet	None	$0.724_{\pm0.0057}$	$\mathbf{0.716_{\pm0.0034}}$	$\mathbf{0.655_{\pm0.0030}}$	$\mathbf{0.635_{\pm0.0086}}$
	ROS	-	$\mathbf{0.698_{\pm0.0037}}$	$0.655_{\pm0.0030}$	$0.619_{\pm0.0084}$
	ROS+	-	$\mathbf{0.713_{\pm0.0036}}$	$\mathbf{0.710_{\pm0.0034}}$	$\mathbf{0.684_{\pm0.0051}}$
RelationNet	None	$0.617_{\pm0.0069}$	$0.614_{\pm0.0039}$	$0.590_{\pm0.0034}$	$0.579_{\pm0.0053}$
	ROS	-	$0.605_{\pm0.0043}$	$0.590_{\pm0.0034}$	$0.573_{\pm0.0052}$
	ROS+	-	$0.605_{\pm0.0040}$	$0.613_{\pm0.0037}$	$0.587_{\pm0.0050}$

Table 6: Results on task-level class imbalance on the UC-Merced dataset.

When looking at the results of task-level imbalance on the UC-Merced dataset (Table 6) we see ProtoNet outperforming the other models in all settings both in the simpler settings and the more complex settings. This could be due to the size of the dataset, perhaps the transfer learning methods are not being exposed to enough information to generalize. The meta-learning algorithms seem to perform well when exposed to less information when compared to the transfer learning methods.

Model	Strategy	5-way-5-shot Train	4-6 shot Linear	1-9 shot Step	1-9 shot Random
Baseline	None	$0.857_{\pm0.0019}$	$\mathbf{0.849_{\pm0.0020}}$	$0.756_{\pm0.0030}$	$0.756_{\pm0.0089}$
	ROS	-	$0.849_{\pm0.0021}$	$0.783_{\pm0.0037}$	$0.782_{\pm0.0068}$
	ROS+	-	$\mathbf{0.859_{\pm0.0019}}$	$0.839_{\pm0.0029}$	$0.827_{\pm0.0040}$
Baseline++	None	$0.785_{\pm0.0033}$	$0.772_{\pm0.0029}$	$0.702_{\pm0.0015}$	$0.687_{\pm0.0077}$
	ROS	-	$0.781_{\pm0.0027}$	$0.758_{\pm0.0029}$	$0.748_{\pm0.0046}$
	ROS+	-	$0.784_{\pm0.0026}$	$0.788_{\pm0.0029}$	$0.768_{\pm0.0038}$
MAML	None	$0.537_{\pm0.0060}$	$0.530_{\pm0.0054}$	$0.513_{\pm0.0058}$	$0.500_{\pm0.0065}$
	ROS	-	$0.538_{\pm0.0060}$	$0.524_{\pm0.0055}$	$0.516_{\pm0.0065}$
	ROS+	-	$0.523_{\pm0.0063}$	$0.521_{\pm0.0065}$	$0.502_{\pm0.0065}$
MatchingNet	None	$0.849_{\pm0.0024}$	$\mathbf{0.849_{\pm0.0020}}$	$\mathbf{0.809_{\pm0.0034}}$	$\mathbf{0.813_{\pm0.0040}}$
	ROS	-	$\mathbf{0.851_{\pm0.0020}}$	$\mathbf{0.837_{\pm0.0032}}$	$\mathbf{0.831_{\pm0.0029}}$
	ROS+	-	$0.849_{\pm0.0022}$	$0.839_{\pm0.0030}$	$\mathbf{0.828_{\pm0.0029}}$
ProtoNet	None	$0.849_{\pm0.0025}$	$0.845_{\pm0.0022}$	$0.767_{\pm0.0033}$	$0.776_{\pm0.0077}$
	ROS	-	$0.835_{\pm0.0024}$	$0.767_{\pm0.0033}$	$0.764_{\pm0.0076}$
	ROS+	-	$0.848_{\pm0.0022}$	$\mathbf{0.841_{\pm0.0027}}$	$0.825_{\pm0.0033}$
RelationNet	None	$0.811_{\pm0.0029}$	$0.810_{\pm0.0026}$	$0.768_{\pm0.0034}$	$0.774_{\pm0.0043}$
	ROS	-	$0.804_{\pm0.0028}$	$0.768_{\pm0.0034}$	$0.769_{\pm0.0044}$
	ROS+	-	$0.801_{\pm0.0028}$	$0.793_{\pm0.0029}$	$0.776_{\pm0.0037}$

Table 7: Results on task-level class imbalance on the WHU-RS19 dataset.

The results on WHU-RS19 (Table 7) are interesting because we see that Baseline, MatchingNet, and ProtoNet are top performers in the standard setting, however, we once again see the impact of task-level imbalance on the transfer learning methods when no over-sampling strategies are applied. The same cannot be said for the meta-learning methods as the impact of the task-level imbalance is less pronounced. That being said, when providing the model with an approach to cope with the imbalance, Baseline becomes competitive with the other models.

Model	Strategy	5-way-5-shot Train	4-6 shot Linear	1-9 shot Step	1-9 shot Random
Baseline	None	$0.868_{\pm 0.0034}$	$\mathbf{0.851}_{\pm \mathbf{0.0043}}$	$\mathbf{0.766}_{\pm \mathbf{0.0044}}$	$0.740_{\pm 0.0075}$
	ROS	-	$\mathbf{0.855}_{\pm \mathbf{0.0043}}$	$0.777_{\pm 0.0050}$	$0.760_{\pm 0.0120}$
	ROS+	-	$\mathbf{0.862}_{\pm \mathbf{0.0038}}$	$\mathbf{0.848}_{\pm \mathbf{0.0049}}$	$\mathbf{0.833}_{\pm \mathbf{0.0060}}$
Baseline++	None	$0.812_{\pm 0.0055}$	$0.798_{\pm 0.0063}$	$0.714_{\pm 0.0037}$	$0.704_{\pm 0.0120}$
	ROS	-	$0.810_{\pm 0.0057}$	$0.786_{\pm 0.0063}$	$\mathbf{0.781}_{\pm \mathbf{0.0078}}$
	ROS+	-	$0.811_{\pm 0.0055}$	$0.816_{\pm 0.0057}$	$0.800_{\pm 0.0066}$
MAML	None	$0.568_{\pm 0.0110}$	$0.567_{\pm 0.0110}$	$0.536_{\pm 0.0096}$	$0.516_{\pm 0.0110}$
	ROS	-	$0.571_{\pm 0.0110}$	$0.571_{\pm 0.0110}$	$0.552_{\pm 0.0120}$
	ROS+	-	$0.556_{\pm 0.0120}$	$0.562_{\pm 0.0120}$	$0.528_{\pm 0.0130}$
MatchingNet	None	$0.798_{\pm 0.0063}$	$0.794_{\pm 0.0066}$	$0.758_{\pm 0.0076}$	$0.763_{\pm 0.0080}$
	ROS	-	$0.796_{\pm 0.0063}$	$\mathbf{0.790}_{\pm \mathbf{0.0067}}$	$0.779_{\pm 0.0069}$
	ROS+	-	$0.794_{\pm 0.0065}$	$0.788_{\pm 0.0067}$	$0.774_{\pm 0.0073}$
ProtoNet	None	$0.811_{\pm 0.0055}$	$0.802_{\pm 0.0057}$	$0.729_{\pm 0.0054}$	$0.721_{\pm 0.0120}$
	ROS	-	$0.788_{\pm 0.0066}$	$0.729_{\pm 0.0054}$	$0.705_{\pm 0.0130}$
	ROS+	-	$0.803_{\pm 0.0060}$	$0.792_{\pm 0.0061}$	$0.772_{\pm 0.0083}$
RelationNet	None	$0.817_{\pm 0.0054}$	$0.814_{\pm 0.0056}$	$0.754_{\pm 0.0061}$	$\mathbf{0.762}_{\pm \mathbf{0.0092}}$
	ROS	-	$0.806_{\pm 0.0059}$	$0.754_{\pm 0.0061}$	$0.754_{\pm 0.0094}$
	ROS+	-	$0.811_{\pm 0.0060}$	$0.792_{\pm 0.0065}$	$0.792_{\pm 0.0075}$

Table 8: Results on task-level class imbalance on the PatternNet dataset.

The results of the task-level imbalance experiments on the PatternNet dataset in Table 8, once again show that the meta-learning methods (excluding MAML) cope better with more complex task-level imbalance when no over-sampling strategy is applied, suggesting they are more robust in terms of being able to cope with more complex imbalanced settings. when dealing with imbalanced tasks. When over-sampling strategies are applied, we see that Baseline and Baseline++ are better suited for dealing with task-level imbalances especially when ROS+ is applied.

Model	Strategy	5-way-5-shot Train	4-6 shot Linear	1-9 shot Step	1-9 shot Random
Baseline	None	$0.710_{\pm 0.0041}$	$0.694_{\pm 0.0040}$	$0.657_{\pm 0.0029}$	$0.609_{\pm 0.0096}$
	ROS	-	$\mathbf{0.699}_{\pm \mathbf{0.0042}}$	$0.669_{\pm 0.0029}$	$0.623_{\pm 0.0090}$
	ROS+	-	$\mathbf{0.709}_{\pm \mathbf{0.0040}}$	$\mathbf{0.708}_{\pm \mathbf{0.0033}}$	$0.673_{\pm 0.0063}$
Baseline++	None	$0.638_{\pm 0.0034}$	$0.621_{\pm 0.0034}$	$0.587_{\pm 0.0032}$	$0.538_{\pm 0.0069}$
	ROS	-	$0.633_{\pm 0.0032}$	$0.623_{\pm 0.0031}$	$0.601_{\pm 0.0049}$
	ROS+	-	$0.633_{\pm 0.0031}$	$0.643_{\pm 0.0033}$	$0.623_{\pm 0.0041}$
MAML	None	$0.501_{\pm 0.0054}$	$0.503_{\pm 0.0044}$	$0.480_{\pm 0.0046}$	$0.452_{\pm 0.0062}$
	ROS	-	$0.502_{\pm 0.0050}$	$0.503_{\pm 0.0290}$	$0.475_{\pm 0.0059}$
	ROS+	-	$0.493_{\pm 0.0052}$	$0.498_{\pm 0.0052}$	$0.467_{\pm 0.0059}$
MatchingNet	None	$0.690_{\pm 0.0038}$	$0.681_{\pm 0.0040}$	$\mathbf{0.659}_{\pm \mathbf{0.0034}}$	$\mathbf{0.647}_{\pm \mathbf{0.0057}}$
	ROS	-	$0.683_{\pm 0.0039}$	$\mathbf{0.680}_{\pm \mathbf{0.0037}}$	$\mathbf{0.662}_{\pm \mathbf{0.0049}}$
	ROS+	-	$0.684_{\pm 0.0038}$	$0.681_{\pm 0.0037}$	$0.665_{\pm 0.0049}$
ProtoNet	None	$0.713_{\pm 0.0036}$	$\mathbf{0.706}_{\pm \mathbf{0.0033}}$	$0.648_{\pm 0.0027}$	$0.629_{\pm 0.0077}$
	ROS	-	$0.690_{\pm 0.0034}$	$0.648_{\pm 0.0027}$	$0.613_{\pm 0.0077}$
	ROS+	-	$0.708_{\pm 0.0034}$	$0.700_{\pm 0.0033}$	$\mathbf{0.675}_{\pm \mathbf{0.0048}}$
RelationNet	None	$0.684_{\pm 0.0035}$	$0.678_{\pm 0.0034}$	$0.635_{\pm 0.0028}$	$0.627_{\pm 0.0064}$
	ROS	-	$0.669_{\pm 0.0037}$	$0.635_{\pm 0.0029}$	$0.619_{\pm 0.0062}$
	ROS+	-	$0.671_{\pm 0.0035}$	$0.663_{\pm 0.0035}$	$0.660_{\pm 0.0047}$

Table 9: Results on task-level class imbalance on the AID dataset.

In general, the meta-learning models (excluding MAML) cope better with more complex task-level imbalance when no over-sampling strategy is applied, suggesting they are more robust when dealing with imbalanced tasks. When over-sampling strategies are applied, we see that Baseline and ProtoNet are better suited for dealing with task-level imbalances especially when ROS+ is applied. While the results of Table 9 present the task-level imbalance, we are looking at the results of a combined imbalance experiment.

The results from Tables 5 - 9 suggest that the Baseline transfer learning method is less resistant to class-imbalanced tasks without the support of oversampling strategies when compared to meta-learners such as ProtoNet and MatchingNet. While a select few meta-learning algorithms (ProtoNet and MatchingNet) are more resistant to class-imbalanced tasks when no strategies are applied, we see that Baseline becomes very competitive when ROS and ROS+ are applied to the underlying data. With random oversampling being a very simple methods for tacking class imbalance, we rank the models based on their performance when ROS+ is applied. We see that the Baseline transfer learning methods ranks first in performance followed by meta-learning methods ProtoNet and MatchingNet. Based on the ranking and the simplicity of random over-sampling as a strategy to cope with class imbalance we propose that the Baseline transfer learning method is a suitable approach for tackling the class imbalance problem in the domain of remote sensing.

7 Conclusion and Future Work

7.1 Conclusion

In this thesis, a structural comparison of various few-shot learning methods was formulated in a remote sensing setting, focusing on comparing transfer learning and meta-learning algorithms. More specifically, we compare 2 well established transfer-learning methods against 4 well established meta-learning methods across 5 datasets of varying difficulty under different experimental settings namely, in a standard setting, network-depth setting, cross-domain setting, and class imbalance setting. All of these are realistic scenarios for researchers who are responsible for training and deploying models into production. With the complexity of remote sensing imagery along with the need for robust model performance under different circumstances, it is crucial to select the appropriate model for reliable and accurate outcomes. The findings provide a guideline for model selection along with some details about the strengths and weaknesses of the different models in varying settings.

With the vast majority of previous works exploring few-shot learning in the remote sensing domain advocating for complex meta-learning algorithms, our initial hypothesis states that the standard transfer learning methods are robust and competitive enough to be deployed in the majority of settings. Based on the rankings of the methods across the various experiments, we can conclude that in general Baseline transfer learning is the most robust approach to few-shot learning in the remote sensing domain except when training on one dataset and testing on another. In a cross-domain setting, it is suggested to use meta-learning algorithms MatchingNet or ProtoNet and in a low-shot setting, it is suggested to use either Baseline++ transfer learning method or MatchingNet. While the network depth experiments are performed using varying architectures, namely convolutional neural networks for the shallow backbones and residual networks for the deeper backbones, the results are still relevant as the purpose of the residual networks are to overcome issues faces by deeper architectures (vanishing/exploding gradient). Our results are in line with other previous works comparing the performance of complex meta-learners to transfer learners in a few-shot learning setting. Huisman et al. [28] also conclude that meta-learning algorithms, namely MAML and Reptile are successful in low-data settings (i.e., 1-shot), however, the features learned by transfer learning models are more diverse and generalize better. Chen et al. [5] run similar experiments in the domain of natural scene imagery and conclude that the baseline transfer learning models outperform the complex meta-learning models in most experiment settings. They also show that in a low-data setting, meta-learning algorithms are more successful when compared to transfer learning models. This suggests that in the low-data setting (1-shot), the transfer learning models aren't exposed to enough information to generalize. However, we see that the Baseline++ transfer learning method is competitive when compared to the complete meta-learners.

Few-shot learners as a technology have the power to contribute to the remote sensing domain, contributing to better decision-making and resource management on a larger scale. When considering the volume and veracity of the data at hand, it is suggested to use Baseline transfer learning as the model is less volatile during training on larger backbones and when faced with class imbalance the data-level techniques help in contributing to improved model performance. Simple yet effective methods for tackling the challenges that researchers face in the domain of remote sensing imagery.

7.2 Future Work

With the endless combination of possible experiment combinations, further research exploring the impact of cross-domain training on the few-shot learning models is in order. More specifically, it would make sense to consider cross-domain experiments with more complex backbone networks. The class imbalance and low-shot setting experiments would also benefit from experiments run with more complex backbone networks, to draw more significant conclusions on the impact of network depth. We see that network depth has a clear impact on model performance, hence it would be beneficial to explore the impact of network depth in the low-shot, cross-domain, and class imbalance settings. While we have motivate the use of data-level approaches to tackling class imbalance, it may also be beneficial for future research to explore how algorithmic-level approaches for tackling class imbalance have an impact on tasks in the domain of remote sensing imagery.

Continuous learning [53], which refers to a model's ability to retain information when faced with new tasks is also relevant in the domain of remote sensing. Exploring which models can retain information while continuing to learn would be beneficial to remote sensing practitioner, as it would allow them to continuously reuse and grow single models for specific tasks instead of maintaining multiple models.

Future works focusing on the application of few-shot learning methods on other remote sensing tasks such as image segmentation, object detection, and multi-label learning would be relevant to provide a broader overview of which models are suitable in the remote sensing domain as they have their own unique set of challenges. Jakubik et al. [29] have released a foundation model for Earth Observation that utilizes pre-training and fine-tuning, focusing on data originating for the United States. They utilize the foundation model on several downstream tasks including imputation and segmentation. Exploring the impact of domain adaptation and network depth on their foundation model could also prove useful for remote sensing practitioners, providing a general approach to several downstream tasks beyond classification with global reach.

A Verification of Implementation

In order to verify the implementation, some initial runs were performed on the same datasets to ensure that the various settings were running correctly.

Model	**Acc (CLFSL)**	**Acc (CIFSL)**	**Acc (Ours)**
Baseline	$0.625_{\pm 0.0069}$	$0.626_{\pm 0.0070}$	$0.615_{\pm 0.0068}$
Baseline++	$0.664_{\pm 0.0066}$	$0.664_{\pm 0.0066}$	$0.636_{\pm 0.0068}$
MAML	$0.627_{\pm 0.0071}$	$0.618_{\pm 0.0071}$	$0.623_{\pm 0.0069}$
ProtoNet	$0.642_{\pm 0.0072}$	$0.643_{\pm 0.0071}$	$0.606_{\pm 0.0072}$
RelationNet	$0.666_{\pm 0.0069}$	$0.647_{\pm 0.0068}$	$0.618_{\pm 0.0068}$
MatchingNet	$0.634_{\pm 0.0066}$	$0.622_{\pm 0.0069}$	$0.607_{\pm 0.0073}$

Table 10: The results from standard 5-way-5-shot experiments using the Mini-ImageNet dataset [70].

References

[1] Najd Alosaimi, Haikel Salem Alhichri, Yakoub Bazi, Belgacem Ben Youssef, and Naif A. Alajlan. Self-supervised learning for remote sensing scene classification under the few shot scenario. *Scientific Reports*, 13, 2023.

[2] Antreas Antoniou, Harrison Edwards, and Amos J. Storkey. How to train your MAML. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[3] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[4] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *CoRR*, abs/1904.04232, 2019.

[5] Xiliang Chen, Guobin Zhu, Mingqing Liu, and Zhaotong Chen. Few-shot remote sensing image scene classification based on multiscale covariance metric network (mcmnet). *Neural Networks*, 163: 132–145, 2023.

[6] Xinshi Chen, Hanjun Dai, Yu Li, Xin Gao, and Le Song. Learning to stop while learning to predict. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1520–1530. Proceedings of Machine Learning Research, 2020.

[7] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

[8] Gong Cheng, Xingxing Xie, Junwei Han, Lei Guo, and Gui-Song Xia. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3735–3756, 2020.

[9] Arkabandhu Chowdhury, Mingchao Jiang, Swarat Chaudhuri, and Chris Jermaine. Few-shot image classification: Just use a library of pre-trained feature extractors and a simple classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9425–9434. IEEE, 2021.

[10] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9268–9277. Computer Vision Foundation / IEEE, 2019.

[11] Sara del Río, José Manuel Benítez, and Francisco Herrera. Analysis of data preprocessing increasing the oversampling ratio for extremely imbalanced big data classification. In *2015 IEEE TrustCom/BigDataSE/ISPA, Helsinki, Finland, August 20-22, 2015, Volume 2*, pages 180–185. IEEE, 2015.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE Computer Society, 2009.

[13] Vincent Dumoulin, Neil Houlsby, Utku Evci, Xiaohua Zhai, Ross Goroshin, Sylvain Gelly, and Hugo Larochelle. Comparing transfer and meta learning approaches on a unified few-shot classification benchmark. *CoRR*, abs/2104.02638, 2021.

[14] Li Fei-Fei, Robert Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006.

[15] Alberto Fernández, Sara del Río, N. Chawla, and Francisco Herrera. An insight into imbalanced big data classification: outcomes and challenges. *Complex and Intelligent Systems*, 3:105–120, 2017.

[16] Michael Fink. Object classification from a single example utilizing class relevance metrics. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, pages 449–456, 2004.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017.

[18] Jie Geng, Bohan Xue, and Wen Jiang. Foreground-background contrastive learning for few-shot remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–12, 2023.

[19] Hassan Gharoun, Fereshteh Momenifar, Fang Chen, and Amir H. Gandomi. Meta-learning approaches for few-shot learning: A survey of recent advances. *CoRR*, abs/2303.07502, 2023.

[20] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 4367–4375. Computer Vision Foundation / IEEE Computer Society, 2018.

[21] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. *Deep Learning*. Adaptive computation and machine learning. MIT Press, 2016. ISBN 978-0-262-03561-3.

[22] Jian Guan, Jiabei Liu, Jianguo Sun, Pengming Feng, Tong Shuai, and Wenwu Wang. Meta metric learning for highly imbalanced aerial scene classification. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 4047–4051. IEEE, 2020.

[23] Yunhui Guo, Noel Codella, Leonid Karlinsky, James V. Codella, John R. Smith, Kate Saenko, Tajana Rosing, and Rogério Feris. A broader study of cross-domain few-shot learning. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVII*, volume 12372 of *Lecture Notes in Computer Science*, pages 124–141. Springer, 2020.

[24] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE Computer Society, 2016.

[26] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[27] Mike Huisman, Jan N. van Rijn, and Aske Plaat. A survey of deep meta-learning. *Artificial Intelligence Review*, 54(6):4483–4541, 2021.

[28] Mike Huisman, Aske Plaat, and Jan N. van Rijn. Understanding transfer learning and gradient-based meta-learning techniques. *Mach. Learn.*, 113(7):4113–4132, 2024.

[29] Johannes Jakubik, Sujit Roy, C. E. Phillips, Paolo Fraccaro, Denys Godwin, Bianca Zadrozny, Daniela Szwarcman, Carlos Gomes, Gabby Nyirjesy, Blair Edwards, Daiki Kimura, Naomi Simumba, Linsong Chu, S. Karthik Mukkavilli, Devyani Lambhate, Kamal Das, Ranjini Bangalore, Dário A. B. Oliveira, Michal Muszynski, Kumar Ankur, Muthukumaran Ramasubramanian, Iksha Gurung, Sam Khallaghi, Hanxi Li, Michael Cecil, Maryam Ahmadi, Fatemeh Kordi, Hamed Alemohammad, Manil Maskey, Raghu K. Ganti, Kommy Weldemariam, and Rahul Ramachandran. Foundation models for generalist geospatial artificial intelligence. *CoRR*, abs/2310.18660, 2023.

[30] Justin M. Johnson and Taghi M. Khoshgoftaar. Survey on deep learning with class imbalance. *J. Big Data*, 6:27, 2019.

[31] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8419–8428. IEEE, 2019.

[32] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part V*, volume 12350 of *Lecture Notes in Computer Science*, pages 491–507. Springer, 2020.

[33] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232, 2016.

[34] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, 2017.

[35] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science (New York, N.Y.)*, 350(6266):1332—1338, 2015.

[36] Haebeom Lee, Hayeon Lee, Donghyun Na, Saehoon Kim, Minseop Park, Eunho Yang, and Sung Ju Hwang. Learning to balance: Bayesian meta-learning for imbalanced and out-of-distribution tasks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[37] Joffrey L. Leevy, Taghi M. Khoshgoftaar, Richard A. Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5:42, 2018.

[38] Lingjun Li, Junwei Han, Xiwen Yao, Gong Cheng, and Lei Guo. Dla-matchnet for few-shot remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9): 7844–7853, 2021.

[39] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007. IEEE Computer Society, 2017.

[40] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *Eureopean Conference on Computer Vision (ECCV)*, volume 9905 of *Lecture Notes in Computer Science*, pages 21–37. Springer, 2016.

[41] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2537–2546. Computer Vision Foundation / IEEE, 2019.

[42] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3431–3440. IEEE Computer Society, 2015.

[43] Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofei Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152:166–177, 2019.

[44] Daniela Massiceti, Luisa M. Zintgraf, John Bronskill, Lida Theodorou, Matthew Tobias Harris, Edward Cutrell, Cecily Morrison, Katja Hofmann, and Simone Stumpf. ORBIT: A real-world few-shot dataset for teachable object recognition. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 10798–10808. IEEE, 2021.

[45] Saeid Motiian, Quinn Jones, Seyed Mehdi Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6670–6680, 2017.

[46] Maxim Neumann, André Susano Pinto, Xiaohua Zhai, and Neil Houlsby. Training general representations for remote sensing using in-domain knowledge. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 6730–6733. IEEE, 2020.

[47] Mateusz Ochal, Jose Vazquez, Yvan R. Petillot, and Sen Wang. A comparison of few-shot learning methods for underwater optical and sonar image classification. *CoRR*, abs/2005.04621, 2020.

[48] Mateusz Ochal, Massimiliano Patacchiola, Jose Vazquez, Amos J. Storkey, and Sen Wang. Few-shot learning with class imbalance. *IEEE Trans. Artif. Intell.*, 4(5):1348–1358, 2023.

[49] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[50] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035. Curran Associates, Inc., 2019.

[51] Otávio Augusto Bizetto Penatti, Keiller Nogueira, and Jefersson Alex dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 44–51. IEEE Computer Society, 2015.

[52] Lam Pham, Khoa Tran, Dat Ngo, Jasmin Lampert, and Alexander Schindler. Remote sensing image classification using transfer learning and attention based deep neural network. *CoRR*, abs/2206.13392, 2022.

[53] Preston Billion Polak, Joseph D. Prusa, and Taghi M. Khoshgoftaar. Low-shot learning and class imbalance: a survey. *J. Big Data*, 11(1):1, 2024.

[54] Hang Qi, Matthew Brown, and David G. Lowe. Low-shot learning with imprinted weights. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5822–5830. Computer Vision Foundation / IEEE Computer Society, 2018.

[55] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2017.

[56] William J Reed. The pareto, zipf and other power laws. *Economics Letters*, 74(1):15–19, 2001.

[57] Marc Rußwurm, Sherrie Wang, Marco Körner, and David B. Lobell. Meta-learning for few-shot land cover classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 788–796. Computer Vision Foundation / IEEE, 2020.

[58] Nelly Rosaura Palacios Salinas, Mitra Baratchi, Jan N. van Rijn, and Andreas Vollrath. Automated machine learning for satellite data: Integrating remote sensing pre-trained models into automl systems. In Yuxiao Dong, Nicolas Kourtellis, Barbara Hammer, and José Antonio Lozano, editors, *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track - European Conference, ECML PKDD 2021, Bilbao, Spain, September 13-17, 2021, Proceedings, Part V*, volume 12979 of *Lecture Notes in Computer Science*, pages 447–462. Springer, 2021.

[59] Dvir Samuel, Yuval Atzmon, and Gal Chechik. From generalized zero-shot learning to long-tail with class descriptors. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*, pages 286–295. IEEE, 2021.

[60] Amrith Setlur, Oscar Li, and Virginia Smith. Two sides of meta-learning evaluation: In vs. out of distribution. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 3770–3783, 2021.

[61] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*, pages 4077–4087, 2017.

[62] Yisheng Song, Ting Wang, Puyu Cai, Subrota K. Mondal, and Jyoti Prakash Sahoo. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Comput. Surv.*, 55(13s):271:1–271:40, 2023.

[63] Adam J. Stewart, Caleb Robinson, Isaac A. Corley, Anthony Ortiz, Juan M. Lavista Ferres, and Arindam Banerjee. Torchgeo: deep learning with geospatial data. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '22, pages 19:1–19:12. ACM, 2022.

[64] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 403–412. Computer Vision Foundation / IEEE, 2019.

[65] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1199–1208. Computer Vision Foundation / IEEE Computer Society, 2018.

[66] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, pages 266–282. Springer, 2020.

[67] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[68] Ihsan Ullah, Dustin Carrión-Ojeda, Sergio Escalera, Isabelle Guyon, Mike Huisman, Felix Mohr, Jan N. van Rijn, Haozhe Sun, Joaquin Vanschoren, and Phan Anh Vu. Meta-album: Multi-domain meta-dataset for few-shot image classification. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[69] Matt van den Nieuwenhuijzen, Carola Doerr, Henry Gouk, and Jan N. van Rijn. Selecting pre-trained models for transfer learning with data-centric meta-features. In *AutoML Conference 2024 (Workshop Track)*, 2024.

[70] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 3630–3638. Curran Associates Inc., 2016.

[71] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[72] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9196–9205. IEEE, 2019.

[73] Tao Wang, Xiaopeng Zhang, Li Yuan, and Jiashi Feng. Few-shot adaptive faster R-CNN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7173–7182. Computer Vision Foundation / IEEE, 2019.

[74] Gui-Song Xia, Wen Yang, Julie Delon, Yann Gousseau, Hong Sun, and Henri Maître. Structural High-resolution Satellite Image Indexing. In *ISPRS TC VII Symposium - 100 Years ISPRS*, volume XXXVIII, pages 298–303, 2010.

[75] Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3965–3981, 2017.

[76] Yi Yang and Shawn D. Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *18th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, ACM-GIS 2010, November 3-5, 2010, San Jose, CA, USA, Proceedings*, pages 270–279. ACM, 2010.

[77] Zhengwu Yuan, Chan Tang, Aixia Yang, Wendong Huang, and Wang Chen. Few-shot remote sensing image scene classification based on metric learning and local descriptors. *Remote. Sens.*, 15(3):831, 2023.

[78] Qingjie Zeng and Jie Geng. Task-specific contrastive learning for few-shot remote sensing image scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 191:143–154, 2022.

[79] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. Metagan: An adversarial approach to few-shot learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NeurIPS)*, pages 2371–2380, 2018.

[80] Weixun Zhou, Shawn D. Newsam, Congmin Li, and Zhenfeng Shao. Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *CoRR*, abs/1706.03424, 2017.