ISPRS Journal of Photogrammetry and Remote Sensing 195 (2023) 250–268

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Benchmarking and scaling of deep learning models for land cover image classification

Ioannis Papoutsis [a,*], Nikolaos Ioannis Bountos [a,b], Angelos Zavras [a,b], Dimitrios Michail [b], Christos Tryfonopoulos [c]

[a] Institute of Astronomy, Astrophysics, Space Applications & Remote Sensing, National Observatory of Athens, Greece
[b] Department of Informatics & Telematics, Harokopio University of Athens, Greece
[c] Department of Informatics & Telecommunications, University of the Peloponnese, Greece

ARTICLE INFO

Keywords:
Benchmark
Land use land cover image classification
BigEarthNet
Wide Residual Networks
EfficientNet
Deep learning
Model zoo
Transfer learning

ABSTRACT

The availability of the sheer volume of Copernicus Sentinel-2 imagery has created new opportunities for exploiting deep learning methods for land use land cover (LULC) image classification at large scales. However, an extensive set of benchmark experiments is currently lacking, i.e. deep learning models tested on the same dataset, with a common and consistent set of metrics, and in the same hardware. In this work, we use the BigEarthNet Sentinel-2 multispectral dataset to benchmark for the first time different state-of-the-art deep learning models for the multi-label, multi-class LULC image classification problem, contributing with an exhaustive zoo of 62 trained models. Our benchmark includes standard Convolution Neural Network architectures, as well as non-convolutional methods, such as Multi-Layer Perceptrons and Vision Transformers. We put to the test EfficientNets and Wide Residual Networks (WRN) architectures, and leverage classification accuracy, training time and inference rate. Furthermore, we propose to use the EfficientNet framework for the compound scaling of a lightweight WRN, by varying network depth, width, and input data resolution. Enhanced with an Efficient Channel Attention mechanism, our scaled lightweight model emerged as the new state-of-the-art. It achieves 4.5% higher averaged F-Score classification accuracy for all 19 LULC classes compared to a standard ResNet50 baseline model, with an order of magnitude less trainable parameters. We provide access to all trained models, along with our code for distributed training on multiple GPU nodes. This model zoo of pre-trained encoders can be used for transfer learning and rapid prototyping in different remote sensing tasks that use Sentinel-2 data, instead of exploiting backbone models trained with data from a different domain, e.g., from ImageNet. We validate their suitability for transfer learning in different datasets of diverse volumes. Our top-performing WRN achieves state-of-the-art performance (71.1% F-Score) on the SEN12MS dataset while being exposed to only a small fraction of the training dataset.

1. Introduction

The Copernicus program is believed to be a game changer for Earth Observation (EO) science. Free and open data available at this scale, frequency, and quality constitute a fundamental paradigm change in EO (Koubarakis et al., 2019). Today, Copernicus is producing 20 terabytes of satellite data every day, however, the availability of the sheer volume of Copernicus data outstrips our capacity to extract meaningful information. Motivated by the success of deep learning (DL) methods in various data-intensive tasks e.g., in medicine (Esteva et al., 2019), self-driving cars (Maqueda et al., 2018), image classification (Perez and Wang, 2017), machine translation (Vaswani et al., 2018), etc., the remote sensing community has been exploiting deep learning methods to propel the research and development of new applications at scale (Zhu et al., 2017).

One prominent application for remote sensing (RS) imagery is land use/land cover (LULC) classification. Research has focused on both pixel-based (Khatami et al., 2016) and object-based (Qian et al., 2015) approaches. LULC mapping scale may vary from high-resolution (Tong et al., 2020b) to global scale, e.g. the Copernicus Global Land Cover (Buchhorn et al., 2020). Machine learning (Talukdar et al., 2020), and DL (Kussul et al., 2017) methods, in particular, have been widely adopted by the community to classify LULC, with some studies exploiting the multi-temporal nature of satellite data, e.g. Ienco et al. (2017). A particular problem family is multi-label LULC scene cat-

* Corresponding author.
E-mail addresses: ipapoutsis@noa.gr, ipapouts@gmail.com (I. Papoutsis).

https://doi.org/10.1016/j.isprsjprs.2022.11.012
Received 23 January 2022; Received in revised form 8 July 2022; Accepted 18 November 2022
Available online 7 December 2022

egorization (Stivaktakis et al., 2019a). The objective of image scene classification and retrieval is to automatically assign class labels to each RS image scene in an archive, and differs from semantic segmentation tasks for LULC mapping and classification. Adopting DL approaches for RS image scene classification problems have shown excellent performance (Lu et al., 2019).

However, DL leads to highly nonlinear, generally overparameterized models (Du et al., 2018) that are prone to overfit. In order to use DL models that generalize well for previously unseen test data, we need to train them with large amounts of input labeled data. The data-hungry nature of modern machine learning has thus been a barrier for its widespread application in geosciences and RS. The lack of curated datasets and pretrained models tailored to RS data has prevented the use of traditional transfer learning approaches for LULC image classification. Therefore, researchers have used models pre-trained on optical datasets (Sumbul et al., 2020), such as ImageNet (Deng et al., 2009), to facilitate the training of new RS models using smaller labeled datasets. This kind of learned knowledge, however, cannot fully transfer on such a different data distribution. Features learned from optical datasets have different characteristics from the ones found in multispectral satellite imagery, therefore in principle, DL models should be trained from scratch to encode this information.

In order to address the problem of the scarcity of labeled data for training DL models for LULC image classification, (Sumbul et al., 2019) created and published BigEarthNet, a large, labeled dataset, which contains single-date Sentinel-2 patches for multi-label, multi-class LULC scene classification. BigEarthNet is a benchmark dataset that consists of 590,326 Sentinel-2 image patches acquired between June 2017 and May 2018 over the 10 European countries, with spectral bands at 10, 20, and 60-meter resolution. Each image patch is annotated by the multiple land-cover classes (i.e., multi-labels) provided by the CORINE Land Cover (CLC) database of the year 2018 (Copernicus, 2018) based on its detailed Level-3 class nomenclature. In order to increase the effectiveness of BigEarthNet, the authors introduced an alternative class-nomenclature to better describe the complex spatial and spectral information content of the Sentinel-2 imagery. The new classes nomenclature consists of 19 LULC classes (Sumbul et al., 2020). Recently, the dataset was enriched with Synthetic Aperture Radar Sentinel-1 patches (Sumbul et al., 2021c).

We identify three major gaps in LULC scene classification with DL, which we address in our work. First is gap is reproducibility, reusability and provenance of the trained models. Currently, an extensive set of benchmark experiments is lacking, i.e. DL models tested on the same dataset, with a common and consistent set of metrics, and in the same hardware. Published works on BigEarthNet (Section 2.1) are fragmented, and in the absence of baseline studies, it is difficult to appreciate which methods work best. Second is the testing and reporting of results on new, state-of-the-art, model architectures that have shown great promise in non-RS, Computer Vision applications. Given the rapid growth of DL research in this field, new families of approaches have emerged, other than traditional Convolutional Neural Networks (CNN), that is worth exploring in RS. Third is accounting for training efficiency and inference time, in addition to classification accuracy, as critical parameters that define overall model performance. This is especially important for training on large datasets, such as on BigEarthNet (~66 Gb, ~0.5 million image patches) or other similar training datasets for LULC classification, e.g. Hong et al. (2021) and Helber et al. (2019). The sheer size of such datasets leads to significant training time overheads, which becomes a bottleneck for testing different model architectures and ideas. Therefore, new methods for efficient training are required to allow researching, engineering and fine-tuning novel DL architectures, including ablation studies and hyperparameter optimization.

To address these gaps we rigorously benchmark DL models using the BigEarthNet dataset, analyzing their overall performance under the light of both speed (training time and inference rate) and model simplicity vis-à-vis LULC image classification accuracy. We investigate standard architectures, such as CNNs, and test novel, non-convolutional methods, such as Multi-Layer Perceptron (MLP) and Vision Transformer (ViT). To the best of our knowledge, it is the first time that ViT and MLP are used to encode multispectral information for LULC, setting-up a challenging state-of-the-art for future methods. ViTs in particular, are inherently data-hungry. Compared to standard CNNs, the lack of inductive bias renders their training from scratch a way more difficult task, and are therefore difficult to be utilized in tasks with small datasets. These models are typically pretrained in large datasets and then finetuned for the task at hand (Dosovitskiy et al., 2020; Steiner et al., 2021). Incorporating them in our model zoo, we finally make them available for exploitation in the remote sensing domain.

In addition, in order to address the requirement for efficiency in training, we explore lightweight architectures with very few parameters compared to typical CNNs. We focus on the use and adaptation of the framework for scaling EfficientNet (Tan and Le, 2019) encoders, and apply it to the order of magnitude more lightweight Wide Residual Network-WRN (Zagoruyko and Komodakis, 2016). Coupled with an implementation for efficient distributed training on 20 GPUs, we are able to experiment with several variations of such scalable models. Our benchmark identifies a set of novel, efficient, models, which are on par or better for most accuracy metrics with other published works, and with considerably fewer trainable parameters and memory requirements at both training and inference. The benchmark concludes with a new WRN model, enhanced with a spatio-spectral attention mechanism, which achieves the best overall performance and sets the new state-of-the-art (SOTA) for LULC image classification on the BigEarthNet.

Our main contributions can be summarized as follows:

- We benchmark 62 DL models for the task of multi-label, multi-class LULC single image classification.
- We provide a DL model zoo based on Sentinel-2 data. The models and the implementation of our framework can be found on the project's github repository.[1] We also provide the first pretrained Vision Transformer and MLP-mixer networks for multispectral Sentinel-2 data.
- We design and scale a new family of models based on Wide Residuals Networks (WRN) that follow the EfficientNet paradigm for scaling. This is the first time that WRN model compound scaling is applied in a remote sensing context and our results show great promise for performance enhancement in satellite image classification tasks.
- We provide the new SOTA on the BigEarthNet dataset in terms of classification accuracy, training time, and inference rate. Our champion model outperforms a baseline ResNet50 model for all 19 LULC classes, achieving 4.5% higher F-score, having an order of magnitude less trainable parameters.
- We show that convolution-free and lightweight architectures (e.g MLPMixer) can have comparable performance with their convolutional counterparts.

2. Related work

2.1. LULC scene classification with BigEarthNet

Recent works have used BigEarthNet to experiment and test DL models for LULC scene classification. In Sumbul and Demir (2020), a multi-attention strategy that utilizes a bidirectional long short-term memory network is adopted to capture and exploit the spectral and spatial information content of RS imagery. A new study by Koßmann et al. (2021) proposes an oversampling method to cope with the BigEarthNet

[1] https://github.com/Orion-AI-Lab/EfficientBigEarthNet

LULC class imbalance, while in Aksoy et al. (2021) the authors propose a consensual collaborative multi-label learning method for harmonizing the BigEarthNet labels. In Kakogeorgiou and Karantzalos (2021) the authors use the DenseNet (Huang et al., 2017) model and test different Explainable Artificial Intelligence methods to interpret model predictions. Finally, in Chaudhuri et al. (2021) a deep representation learning framework on fused BigEarthNet spectral bands is proposed for the same task.

The BigEarthNet dataset has also been used for unsupervised and/or weekly supervised tasks. In Sumbul et al. (2022) a Deep Metric Learning framework is adopted, where a triplet sampling method is proposed to learn quality feature representations towards content-based image retrieval. In Wang et al. (2020a), a U-Net (Ronneberger et al., 2015) image classifier transferred to segmentation is trained with weak labels, outperforming pixel-level algorithms. The authors in Mañas et al. (2021) show that pre-training with contrastive learning on BigEarthNet outperforms ImageNet pretrained models for LULC scene classification, while in Stojnic and Risojevic (2021) contrastive multiview coding is adopted for self-supervised pretraining. Similarly, in Vincenzi et al. (2021) colorization is proposed as a solid pretext task before using BigEarthNet labels for the LULC scene classification downstream task.

2.2. LULC scene classification with other datasets

The work of Maggiori et al. (2016) has been one of the early works on LC classification with high-resolution RS images using transferable deep models. Since then, deep learning has been extensively used for LULC image classification, in different setups and for various datasets. SEN12MS by Schmitt et al. (2019) is a dataset consisting of 180,662 triplets of dual-pol synthetic aperture radar (SAR) image patches, multispectral Sentinel-2 image patches, and MODIS land cover maps as labels. EuroSAT (Helber et al., 2019) is another single label LULC image classification dataset, and is comprised of ten classes with a total of 27,000 labeled and geo-referenced Sentinel-2 multispectral images. Finally, a typical dataset used for deep learning based LULC high resolution RS scene classification is the well-known UC Merced (UCM) dataset by Yang and Newsam (2010). The UCM dataset consists of 21 different labeled classes, but is relatively small in size, with only 100 images per class, which must then be divided between training and validation sets.

To address the challenge of few training samples for DL, Scott et al. (2017) test a transfer learning with fine-tuning approach and a data augmentation strategy tailored specifically for remote sensing imagery for the UCM dataset. Finding efficient data augmentations is also the focus of Stivaktakis et al. (2019a) for the same dataset. Gómez and Meoni (2021) on the other hand, adopt a semisupervised learning approach to deal with label scarcity, which is tested on both the UCM and EuroSAT datasets. The approach is based on a combination of weak and strong data augmentations along with pseudolabeling. A semi-supervised processing chain based on the appropriate selection of labeled samples through a teacher model is proposed by Fan et al. (2020), leveraging the availability of large amounts of unlabeled very high-resolution RS ShenzhenLC city data, for urban LULC image classification. The heterogeneous urban LC types of the city of Southampton, UK and its surrounding environment were used by Zhang et al. (2018), proposing an MLP-CNN ensemble classifier for capturing deep spatial feature representations spectral discriminative information, for aerial imagery classification.

Chaib et al. (2017) use traditional CNN models for feature extraction, while for the classification the authors rely on Support Vector Machines. This simpler approach is tested on the UCM dataset and the Aerial Image dataset (Xia et al., 2017), a large-scale data set for aerial scene classification with more than 10,000 aerial scene images, annotated with 30 classes. Tong et al. (2020b) setup the Gaofen Image Dataset (GID), a large-scale LC annotated dataset with Gaofen-2 (GF-2) satellite images. The authors exploit GID to pretrain standard CNNs models that capture the contextual information contained in different LC types, and propose a domain adaptation strategy by creating and appropriately selecting pseudolabels from a different target domain of unlabeled high resolution RS images. The transferability of their DL models is showcased on several datasets, including Gaofen-2, Gaofen-1, Jilin-1, Ziyuan-3, Sentinel-2 A, and Google Earth platform data. High resolution RS data are also used by Lee et al. (2020), who propose a spectral domain transformation strategy on individual Landsat −8 multi-temporal pixel data, for creating two-dimensional matrices on which CNN models can be applied for LULC classification.

Finally, Zhang et al. (2020) address the issue of input image resolution, similarly to Efficient scaling, and develop an approach to automatically design a pyramid-like scale sequence that is fed to CNN models for aerial digital photography LULC image classification. Learning multiscale deep representations was also proposed by Zhao and Du (2016) for classifying RS images, an approach that was tested for three custom very high resolution RS datasets.

2.3. EfficientNets in remote sensing

EfficientNet is a family of DL models that are scaled to balance network depth, width, and input data resolution to achieve an optimal performance-training time trade-off. Before the EfficientNets came along, the most common way to scale up CNNs was either by one of three dimensions:

- Depth (number of hidden layers) as in He et al. (2016): although deeper networks tend to provide better image classification accuracy, they are also more difficult to train due to the well-known vanishing gradients problem. Accuracy gains quickly diminish beyond a certain depth.
- Width (number of channels/filters) as in Zagoruyko and Komodakis (2016): while easier to train and able to capture fine-grained features, they encounter difficulties in capturing higher-level image content.
- Image resolution (image size) as in Huang et al. (2019): enhanced resolution of the input imagery in principle provides the CNN with more information.

EfficientNets on the other hand perform Compound Scaling, i.e. scale simultaneously all three dimensions, depth, width, and image resolution, while maintaining a balance between all dimensions of the network.

In RS, EfficientNets have lately gained traction, however in most cases they are used as a lightweight CNN backbone without care on how to scale them for the problem at hand. In Alhichri et al. (2021a) for example, the authors engineer a new CNN model that is based on the pretrained EfficientNet-B3 scaled CNN, enhanced with an attention mechanism for RS image classification. EfficientNet-B0 lightweight backbone with a recurrent attention module is employed for the same problem in Liang and Wang (2021). EfficientNet-B0 and its deeper EfficientNet-B3 version are used in Bazi et al. (2019) for fine-tuning pre-trained CNNs, while in Rahhal et al. (2020) EfficientNet is used as a feature extractor coupled with a set of Softmax classifiers for knowledge adaptation across multiple RS sources. Semantic segmentation of high-resolution RS images is addressed with an EfficientNet-B1 model as lightweight network with attention modules in Liu et al. (2020). An EfficientNet with a reduced number of parameters but improved performance is proposed in Shao et al. (2020) for mapping buildings damaged by a wide range of disasters. In Gómez and Meoni (2021), the authors deploy EfficientNet models for semi-supervised multi-spectral scene classification with few labels. Finally, in Tian et al. (2020) EfficientNet-B0 is used as a backbone, mixed with an attention module, for RS object detection.

Model compound scaling with EfficientNets has been sporadically adopted for addressing remote sensing applications. In Charoenchittang et al. (2021), the authors test and report performance for five different scaled versions of EfficientNet (B0-B4), to classify airport buildings

within Google Earth collected and annotated RGB imagery. Wu et al. (2020a) also use Google Earth imagery for an aircraft type recognition task, and apply compound scaling for balancing network width, depth, and resolution and selecting the best performing EfficientNet. In Zhao et al. (2020a) the authors use EfficientNet as a backbone feature encoder to their network for delineating buildings and argue that employing the compound scaling method allows the scaled model to focus on more relevant regions with enhanced object details. Similarly, Chen et al. (2022) scale an Efficient based model for pavement defect detection and classification. They show preference for the EfficientNetB4 model which although has marginally lower detection accuracy with respect to the EfficientNetB5 counterpart, the former has fewer trainable parameters. Finally, Ye et al. (2022) use and scale EfficientDet (Tan et al., 2020) that combines EfficientNet architecture with a bi-directional feature pyramid network for object detection in very high resolution satellite imagery.

2.4. Wide residual networks in remote sensing

WRNs have been used to a lesser extent than EfficientNets to address remote sensing applications. They have been mainly used as part of benchmark studies and compared with traditional CNNs, e.g. as by Chen et al. (2020) for remote sensing image classification. Similarly, Naushad et al. (2021) show that a WRN encoder has superior performance than other CNNs for LULC classification using the EuroSAT dataset (Helber et al., 2019). A variant of WRN was also used by Kang et al. (2021) as an encoder to construct feature embeddings that are then passed onto the nodes of a graph neural network for multi-label remote sensing image classification.

More complex approaches that build on WRNs have also been researched. Ben Hamida et al. (2018) test different 3-D architectures for airborne image classification considering the WRN trade-offs between network width versus depth. Bai et al. (2018) adopt a modified WRN, with wider convolutional channels and fewer network layers, to identify tsunami induced damages in build-up areas from Synthetic Aperture Radar data. Diakogiannis et al. (2020) focus on a semantic segmentation application, split into sequential tasks and addressed with a hierarchical model. The authors perform an ablation study that follows the WRN philosophy for understanding the performance gains, considering both model complexity and training convergence. Khurshid et al. (2020) build on the WRN concept to propose a new residual unit to limit diminishing feature reuse, as indicated by Zagoruyko and Komodakis (2016). Finally, Md. Rafi et al. (2019) work on hyperspectral image classification and propose an attention transfer architecture for domain adaptation between two WRNs.

2.5. Attention modules in remote sensing

Deep learning architectures that incorporate attention modules have been widely used in remote sensing and have shown to provide improvements for different applications, e.g. for image classification, image segmentation, change detection, and object detection. The main variations include spatial, temporal, channel, cross and self-attention networks, while an overview of the main attention mechanism approaches used in RS is provided by Ghaffarian et al. (2021).

To cope with the large receptive fields of traditional CNNs, Ding et al. (2020) propose a complex attention structure, that focuses on both low level features extracted from the early layers of a CNN, and high level features extracted from the late layers. The architecture is able to enrich the semantic information of low-level features by embedding local focus from high-level features and the algorithm is applied for RS image segmentation. Similarly, Tang et al. (2021) propose a spatial and channel attention consistent model to capture both local and global features for RS image classification. For the same task, Wang et al. (2019b) design a recurrent attention mechanism that is able to fit high-level semantic and spatial features into simple representations, managing to accelerate the convergence rate and improve the classification accuracy. Alhichri et al. (2021a) enhance the EfficientNet-B3 encoder with a variation of the Squeeze-and-Excitation attention mechanism (Hu et al., 2018), that we also test in our work, for RS image classification. Squeeze-and-Excitation channel attention mechanism is also applied by Tong et al. (2020a) for the same problem, using a different encoder. Zhao et al. (2020b) propose two simple spatial and channel attention modules, which are able to reduce the impact of many small objects and complex backgrounds on RS image classification.

Wang et al. (2019a) focus on object detection for VHR RS imagery, and develop an architecture comprising of an encoder–decoder model that extracts features at multiple scales, followed by a different, trainable, attention network for each scale. A multi-scale approach is also adopted by Chen and Shi (2020) for RS image change detection, in which multiple spatial–temporal attention networks are trained to capture such dependencies at various scales.

Self-attention approaches have gained traction in RS. Cao et al. (2020) propose a spatial and channel nonparametric self-attention layer, to enhance the semantic information propagated from representative objects, for RS image classification. A self-attention model is designed by Wu et al. (2020b) to reduce the interference of complex backgrounds and to focus on the most salient region of each image, also for RS image classification. Martini et al. (2021) exploit Sentinel-2 time-series and develop a self-attention-based network tailored for domain adaptation for LC and crop classification in different geographic regions. Finally, in contrast to self-attention approaches where the input is a single embedding sequence, cross-attention combines asymmetrically two separate embedding sequences of the same dimension, e.g. as in Cai and Wei (2020) where the authors develop a cross-attention mechanism for hyperspectral data classification, showing improved performance on several popular relevant RS data sets.

3. Models in the benchmark

In this section, we present the DL models that we deploy and benchmark for LULC scene classification. These models are tested by adapting and customizing state-of-the-art DL architectures in Computer Vision, which have not yet been evaluated in remote sensing for the particular task. Fig. 1 summarizes the workflow of the benchmark.

3.1. Vision transformer

Transformers (Vaswani et al., 2017) are a typical example of an architecture that makes the most of attention mechanism. These architectures have been successfully deployed in tasks concerning natural language processing (Devlin et al., 2018). This success has driven the research community to extend the traditional transformer architecture to computer vision. Recently, transformers have been successfully tested for hyperspectral image classification (Zhong et al., 2021). The Vision Transformer (ViT) in Dosovitskiy et al. (2020) processes images as follows. First, it splits the input image in N non-overlapping patches, and each patch (token) is linearly embedded. A class embedding is prepended to the token sequence, while positional embeddings are added to the patch embeddings to ensure positional information is not lost.

In our implementation, we use a fully connected layer for the encodings. The output of this process is the input to a standard Transformer encoder. The classification is based on the prepended class token or a global average pooling of all tokens if the class token was not prepended (Arnab et al., 2021). We examine 5 versions of the ViT architecture. The first four are identical, and we simply vary the patch size. These models will be referred to as ViT/PatchSize e.g ViT/20. They consist of 8 transformer layers with 4 attention heads. Additionally, we examine a ViT with 12 transformer layers, 10 attention heads, and a patch size equal to 20. It will be denoted as ViTM/20.

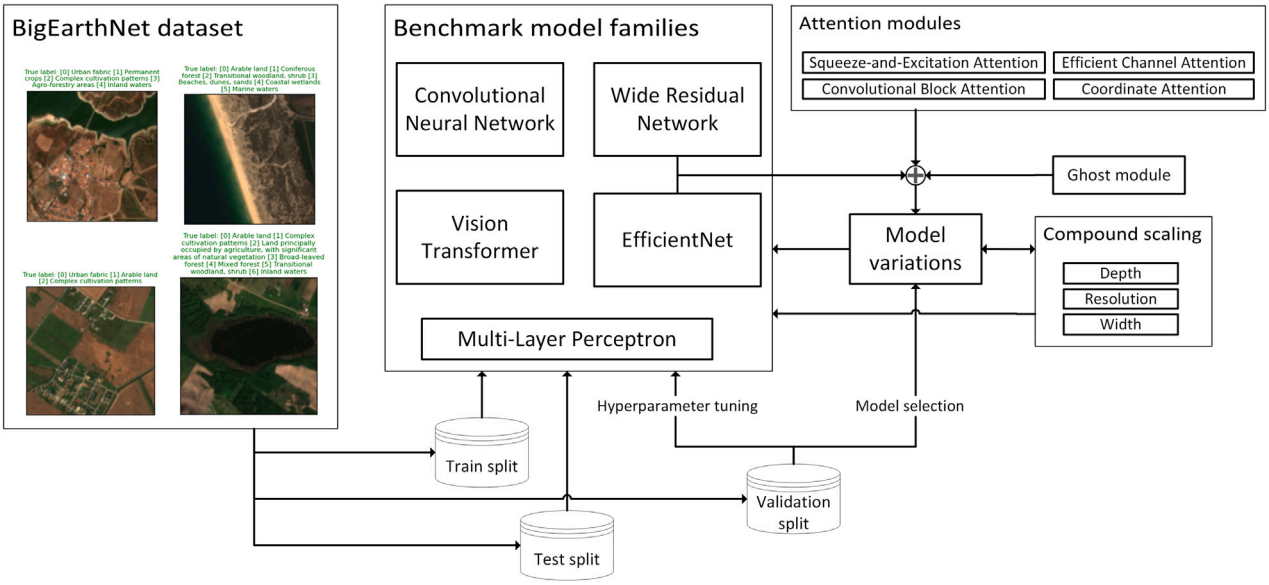

Fig. 1. Workflow for the benchmark. There are five model families tested in the benchmark. For EfficientNets and Wide Residual Networks we test variations with different attention modules and the addition of a Ghost module. We select the best performing model from each of the two base architectures and scale them through compound scaling. EfficientNet and Wide Residual Network architectures are explained in Figs. 2 and 4 respectively.

3.2. Multi-layer perceptrons

Ever since the attention mechanism gained popularity, multiple networks based on attention have emerged. Alternatives to this strategy have made their appearance though, with the reemergence of simple MLP architectures as efficient lightweight models. Recent work in Tolstikhin et al. (2021) has shown that a plain architecture based on MLPs can compete with complex CNN and Transformer architectures. The MLP-Mixer (Tolstikhin et al., 2021) architecture, for example, splits the input in K patches (tokens) and produces an embedding for each one of them. The embeddings are then fed in $\times N$ Mixer Layers. The final prediction is produced by a simple classification head. Each Mixer Layer consists of two blocks: the token-mixing MLP block and the channel-mixing MLP block.

In this work, we build on top of the MLP-Mixer for the fast training of lightweight models with high throughput. We use two versions of the MLP-Mixer in particular. The base version, which we call MLP-Mixer, uses patch size of 12, a hidden dimension of 128 for the linear embeddings, and 4 Mixer layers. Each layer uses channel MLP dimension of 200 and token MLP dimension of 64. The second version called MLPMixerTiny uses a patch size of 6, embedding hidden dimension of 30, and 2 Mixer Layers. The token MLP dimension is set at 12 and the channel MLP dimension at 50. Following 3.1, any MLPMixer variant with different patch size will be referred as MLPMixer/PatchSize.

3.3. EfficientNet and WRN-based models

We deploy in our benchmark the original model in Tan and Le (2019), which introduces a new baseline CNN architecture called EfficientNet-B0. Based on MnasNet (Tan et al., 2019), this baseline network uses the Inverted Residual Block (MBConv Block), as in Howard et al. (2017), a type of residual block used by several mobile-optimized CNNs for efficiency reasons, with the addition of a Squeeze-and-Excitation (SE) block (Hu et al., 2018).

Furthermore, we investigate the impact of yet another efficient CNN encoder, the WRN (Zagoruyko and Komodakis, 2016). Wide Residual Networks constitute an enhancement to the original Deep Residual Networks. Instead of relying on increasing the depth of a Residual Network to improve its accuracy, it was demonstrated that a network could be made shallower and broader without compromising its performance. Prior to the introduction of WRNs, Deep residual networks (e.g., ResNets) were shown to have a fractional boost in performance, but at the cost of roughly doubling the number of layers. This led to the problem of diminishing feature reuse (Srivastava et al., 2015) and overall made the models slower to train. WRNs showed on popular benchmark datasets that having a wider residual network, by widening the ResNet blocks, leads to better performance with respect to deeper counterparts. In Zagoruyko and Komodakis (2016), the authors also show that gradually increasing both depth and width helps until the number of parameters becomes too high and stronger regularization is needed.

We design and deploy two different base models in this benchmark. The first one uses EfficientNet-B0 as a backbone, enhanced with different attention mechanisms and a ghost module that we introduce in this section. The model architecture is presented in Fig. 2. The second base model uses WRN as a backbone and is also tested with different attention mechanisms and with the addition of a ghost module. As our baseline model we use the smallest WRN possible, WRN-10-2, which denotes a residual network that has a depth of 10 convolutional layers and a widening factor of 2. The architecture of the WRN-based family of models is presented in Fig. 4.

3.3.1. EfficientNet/WRN with ghost module

Inspired by the work in Han et al. (2020), we test the effect of a Ghost module for our base architectures. This refers to an essentially standalone replacement layer for standard convolution layers in deep neural network architectures. In principle, this alternative convolutional layer runs linear transformations on fewer feature maps and still fully reveals information of the underlying intrinsic features. The main functionality of the Ghost module we use is to remove redundant copies of unique intrinsic feature maps (Ghost Feature Maps) learned by different convolutional layers in deep networks, so that we preserve the feature-rich representations of the input image while avoiding redundant convolution operations. In this benchmark, we use the suffix -ghost to denote a model that uses a Ghost module in its architecture.

3.3.2. EfficientNet/WRN with attention modules

While conventional CNNs extract features by fusing spatial and channel information within local receptive fields, attention mechanisms can enhance the important parts of the input data either or both in the spatial and the spectral domain. In our benchmark, we experiment

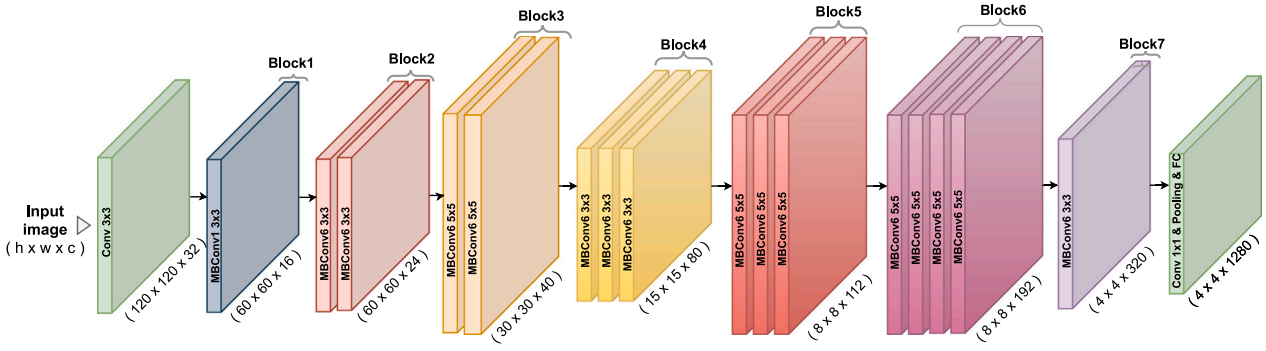

Fig. 2. Our EfficientNet base model architecture implemented in the benchmark. MBConv1 and MBConv6 blocks are explained in Fig. 3.

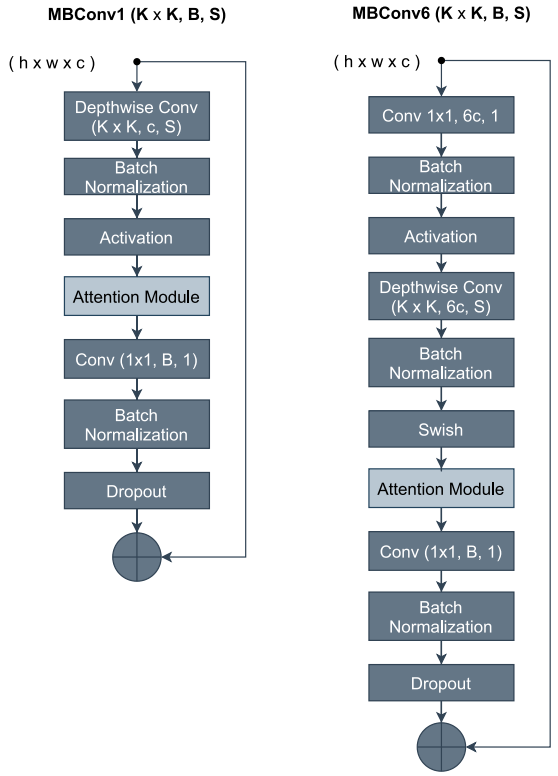

Fig. 3. The modules of MBConv1 and MBConv6 blocks used in the architecture of Fig. 2. With light blue, we mark the position of the Attention mechanism, for which we substitute the different attention modules as explained in Section 3.3. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with different spatial and channel attention mechanisms for both EfficientNet and WRN based models. The exact position of the attention mechanisms is shown in Figs. 3 and 5 for the two base architectures of Figs. 2 and 4 respectively.

Firstly, we evaluate the Squeeze-and-Excitation Attention Module (SE), as in Hu et al. (2018), a channel attention building block, facilitating dynamic channel-wise feature recalibration via channel-wise dependency modeling. Given the input features, the SE block first employs a 2D Global Average Pooling (GAP) for each channel independently, then two fully-connected layers with non-linearity followed by a sigmoid function are used to generate channel weights. The two fully-connected layers are designed to capture non-linear cross-channel interactions, which involves dimensionality reduction for controlling model complexity. Hereinafter, we use the suffix *-SE* to denote a model that uses a Squeeze-and-Excitation Attention module in its architecture.

Channel attention mechanisms, such as the SE, have demonstrated promising results in the design of lightweight mobile networks. Nevertheless, a fundamental shortcoming of those mechanisms is that positional attention information is neglected, which is critical for vision tasks. Later works, such as the Convolutional Block Attention Module (CBAM) (Woo et al., 2018), attempt to exploit positional information with little to no additional computational cost by reducing the channel dimension of the input tensor and then computing spatial attention using convolutions. CBAM consists of two consecutive sub-modules, the Channel Attention Module (CAM) and the Spatial Attention Module (SAM). CAM is similar to the SE with a small modification. Instead of reducing the Feature Maps to a single pixel by GAP, it decomposes the input tensor into 2 subsequent vectors of dimensionality ($c \times 1 \times 1$). One of these vectors is generated by GAP while the other vector is generated by Global Max Pooling (GMP). Average pooling is mainly used for aggregating spatial information, whereas max pooling preserves much richer contextual information in the form of edges of the object within the image, which leads to finer channel attention. The authors validate this in their experiments where they show that using both GAP and GMP gives better results than using just GAP as in the case of SE. SAM, on the other hand, is a three-step sequential procedure. The first phase is called the Channel Pool and it contains max pooling and average pooling operations applied across the channels to the input ($c \times h \times w$), to generate an output with shape ($2 \times h \times w$). This is the input to a convolution layer that outputs a 1-channel feature map ($1 \times h \times w$). After feeding the output into a BatchNorm and an optional ReLU, the data enter a Sigmoid Activation layer. Finally, the attention maps produced by CAM and SAM are multiplied with the input feature map for adaptive feature refinement. In this work, we use the suffix *-CBAM* to denote a model that uses a Convolutional Block Attention Module in its architecture.

CBAM adopts convolutions to capture local relations but fails to model long-range dependencies. In order to deal with this drawback, (Hou et al., 2021) proposed Coordinate Attention (CA) module, a novel attention mechanism which embeds positional information into channel attention so that the network can focus on large important regions at little computational cost. CA captures long-range spatial dependencies while alleviating positional information loss, caused by the 2D global pooling, by factorizing channel attention into two parallel 1D feature encoding processes that effectively integrate spatial coordinate information into the generated attention maps. More precisely, the CA approach uses two 1D global pooling operations to aggregate the input features in the vertical and horizontal directions into two separate direction-aware feature maps. These two feature maps with embedded direction-specific information are then independently encoded into two attention maps, each of which captures long-range dependencies of the input feature map along one spatial direction. As a result, the positional information can be preserved in the generated attention maps. Both attention maps are then applied to the input feature map via multiplication to emphasize the representations of interest. We use the

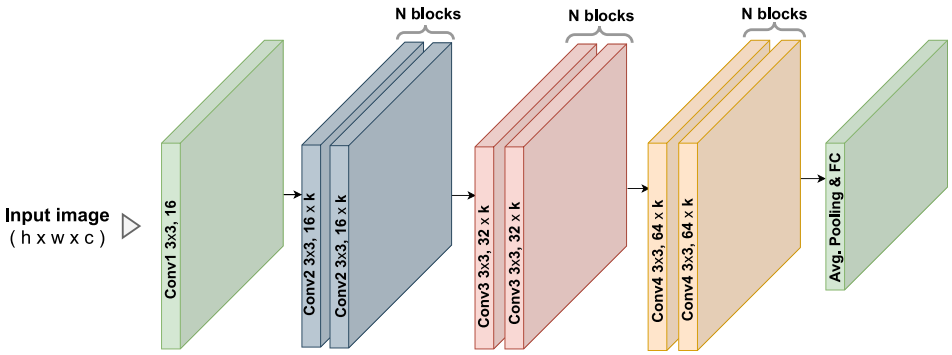

Fig. 4. Our WRN base model architecture implemented in the benchmark. The blocks of the architecture are explained in Fig. 5.

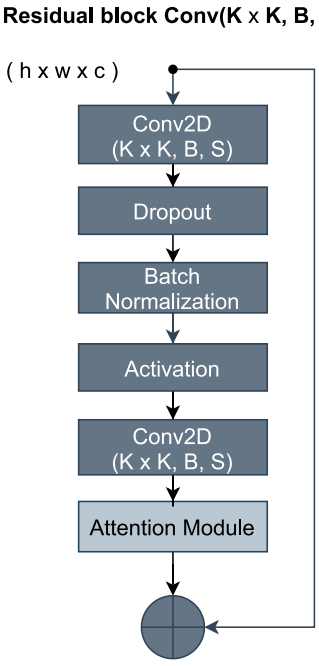

Fig. 5. The modules of the residual block used in the architecture of Fig. 4. With light blue, we mark the position of the Attention mechanism, for which we substitute the different attention modules as explained in Section 3.3.

suffix *-CA* to denote a model that uses a coordinate attention module in its architecture.

Although the strategy of dimensionality reduction for controlling model complexity is widely used in the aforementioned attention modules, Wang et al. (2020b) claim that dimensionality reduction has side effects on channel attention prediction and it is inefficient and unnecessary to capture dependencies across all channels. Therefore, they propose the Efficient Channel Attention (ECA) module, which avoids dimensionality reduction and captures cross-channel interaction in an efficient way, so that both efficiency and effectiveness are preserved. It first performs channel-wise global average pooling and then captures channel attention through a fast $1D$ convolution, whose kernel size is adaptively determined by a non-linear mapping of the channel dimension. The ECA block models local cross-channel interaction by considering every channel and its k neighbors. We use the suffix *-ECA* to denote a model that uses an Efficient Channel Attention module in its architecture.

3.4. Method for scaling-up our EfficientNet and WRN models designs

Our framework for scaling the base models of Figs. 2 and 4, and their variants with the different attention mechanisms and the ghost module, is the compound model scaling method, as in Tan and Le (2019). It consists of a set of rules to scale the three dimensions of our model architectures, depth, width, and input data resolution, using a compound coefficient ϕ. Through that coefficient, dimensions do scale uniformly. ϕ represents how many more resources are available for the model to scale, while α, β, γ are parameters that assign those extra resources to the dimensions of the network:

$$\begin{aligned} &depth \;\; d = a^{\phi} \\ &width \;\; w = \beta^{\phi} \\ &resolution \;\; r = \gamma^{\phi} \\ &s.t. \;\; \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\ &\alpha \geq 1, \beta \geq 1, \gamma \geq 1, \end{aligned} \tag{1}$$

We start from a baseline EfficientNetB0 (Fig. 2) and WRNB0 (Fig. 4) models and scale them in two steps: first we determine the $\alpha, \beta, and \gamma$ coefficients for our EfficientNet and WRN base models. For EfficientNet we use the same coefficients determined by grid-search and used by the authors. For WRN on the other hand, we perform a grid search to determine $\alpha, \beta, and \gamma$. At the same time, inspired by Bello et al. (2021), we evaluate the effect of width scaling prioritization over depth, in contrast to the EfficientNet paradigm. It should be noted that due to the rounding of the number of filters and the number of blocks, some coefficients result in the same network. In such cases, we kept the coefficients resulting in the lowest $\alpha \cdot \beta^2 \cdot \gamma^2$ product, based on Eq. (1). We report the results of our grid search for WRN in Table 1 and summarize the coefficients used for scaling our EfficientNet and WRN base models in Table 2. Second, we fix the aforementioned $\alpha, \beta, and \gamma$ and scale our baseline model by varying ϕ, using Eq. (1). Our baseline B0 models make use of 60 × 60 resolution images, reaching up to 120 × 120 for our B7 models. Eight models can be generated, from EfficientNetB0 up to EfficientNetB7. Similarly, another eight WRN models can be generated, from B0 to B7. To the best of our knowledge, we are the first to use WRNs, enhance them with different attention modules, and scale them using the EfficientNet paradigm.

Table 1
Grid-search used to determine the optimum compound scaling coefficients α, β, γ, which are later used for scaling our WRN model.

α	β	γ	F-score (%)	Training time (hours.mins)	Model size
1.1	1.2	1.1	**75.6**	0.20	433,195
1.2	1.1	1.1	75.2	0.21	373,367
1.2	1.3	1.1	75.5	0.22	520,091
1.3	1.1	1.1	75.1	0.23	410,471
1.4	1.1	1.1	74.8	0.24	447,114
1.1	1.2	1.2	74.6	0.22	433,195
1.2	1.1	1.2	74.2	0.23	373,367
1.1	1.2	1.3	74.4	0.23	433,195
1.2	1.1	1.3	74.0	0.23	373,367

Table 2
Compound scaling coefficients α, β, γ determined by grid-search and used for scaling our EfficientNet and WRN models.

Model	α	β	γ
EfficientNet	1.2	1.1	1.1
WRN	1.1	1.2	1.1

3.5. K-branch CNN

K-Branch CNN, is a variant of the attention based model introduced by Sumbul and Demir (2019). Assuming that different bands come with different spatial resolution, K-Branch CNN processes the bands grouped by spatial resolution in different branches. Each branch produces a local descriptor for the respective spatial resolution. To classify the input image, the respective descriptors are concatenated and fed to a fully connected layer. We have adapted the implementation provided by the authors to our framework to conduct our experiments.

3.6. Traditional convolutional neural networks

In our study, we include three traditional convolutional neural network families to serve as baselines i.e VGG, ResNet and DenseNet. These architectures have been widely used in both remote sensing e.g. Sumbul et al. (2021b), Helber et al. (2019) and Kakogeorgiou and Karantzalos (2021) and computer vision applications. We briefly discuss the core ideas in the following subsections.

3.6.1. VGG

VGG (Simonyan and Zisserman, 2015) is one of the reference convolutional neural networks. It received the first and second place in the ImageNet challenge 2014 in the localization and localization tracks respectively. It is designed to use small 3×3 convolutional filters, while increasing the depth of the neural network.

3.6.2. Residual neural network

Residual Neural Networks (He et al., 2016) have been the golden standard in multiple tasks for a long time. They introduced skip-connection blocks that enabled the training of very deep neural networks, up to 8x deeper than VGG and showed that increased depth can lead to considerable boost in accuracy, earning the first place in the classification track of the ImageNet challenge 2015.

3.6.3. Dense convolutional neural network

In a similar fashion as ResNets, DenseNet (Huang et al., 2017) proposes to connect all layers directly instead of single skip connections. To achieve that, each layer receives as inputs the concatenated features of all preceding layers. A composition of such densely connected blocks with transition layers that perform downsampling (including pooling and convolutions) results to the Dense Convolutional Neural Network.

4. Experiments

4.1. Dataset and experimental setup

We use the BigEarthNet pre-defined splits as in Sumbul et al. (2021c) to train, test, and validate our models, based on the 19 land use classes nomenclature. This corresponds to 295,118 (33 GB), 147,559 (17 GB) and 147,559 (17 GB) Sentinel-2 image patches respectively. In addition to the 10th spectral band of Sentinel-2, which was also excluded in the original BigEarthNet implementation, we also excluded the 1st and 9th spectral bands as they do not contain information regarding the Earth's surface.

We benchmark five model families in this work. First, we test traditional CNN models, including ResNets, DenseNets, and VGG. We then proceed to test the non-convolutional MLP models (Section 3.2). We use two versions of the MLPMixer (Tolstikhin et al., 2021), which we refer to as "MLPMixer" and "MLPMixerTiny", with 446,723 and 40,863 trainable parameters respectively. Transformer networks ViT (Dosovitskiy et al., 2020) are also trained and reported in our models zoo (Section 3.1). We then benchmark EfficientNetB0 baseline network (Tan and Le, 2019) and test the impact of the attention mechanisms (squeeze and excitation, efficient channel attention, coordinate attention and convolutional block attention module) and variations with and without a ghost module, as described in Section 3.3. A total of eight models is produced. Finally, we repeat the same set of experiments for WRNB0 baseline network and its variations. A total of ten models is produced, two more compared to EfficientNetB0, since the latter already contains the squeeze and excitation attention module in its baseline architecture. Based on these experiments, we select the best performing EfficientNetB0 and WRNB0 variation, considering both classification accuracy and training time metrics. These models are subsequently scaled-up from B0 up to B7 using the compound scaling methodology described in Section 3.4.

Given the computing resources available at the HPC infrastructure and depending on the model size, batch size varies between 32 and 256 and the learning rate varies between 0.00001 and 0.001 with a step decay at epochs 24 or 27. We train all models for a total of 30 epochs using the Adam optimizer. The learning rate is scaled by the number of workers in each run. The weights are initialized randomly. We select to minimize the Binary Cross Entropy loss function.

4.2. Evaluation metrics

In supervised learning, for a multi-class problem, different methods are used to evaluate the generalization performance of a model, such as Accuracy and Area Under the Receiver Operating Characteristic (ROC) curve. In a multi-label setting, which is a generalization of multi-class classification, evaluating performance is more complicated than single-label classification problems, due to the simultaneous presence of multiple labels in the scene. Several problem transformation methods exist for multi-label classification. We adopt the binary relevance method (Read et al., 2011), which entails training distinct binary classifiers, one for each label. Each node in the output layer of our networks uses the sigmoid activation, so that a probability of class membership for the label is predicted. The results of each test sample can be assigned to one of the four categories:

- True Positive (TP) - the label is positive and the prediction is also positive
- True Negative (TN) - the label is negative and the prediction is also negative
- False Positive (FP) - the label is negative but the prediction is positive
- False Negative (FN) - the label is positive but the prediction is negative

Here we define a set D of N examples and Y_i to be a family of ground truth label sets and $P_i = h(x_i)$ to be a family of predicted label set. Following the formulation in Zhang and Zhou (2014), the union set of all unique labels is:

$$L = \bigcup_{i=0}^{N-1} L_i \tag{2}$$

While the definition of indicator function I_A on a set A is presented as:

$$I_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Micro Precision (precision averaged over all label pairs) is defined as:

$$Pr_{micro} = \frac{TP}{TP + FP} = \frac{\sum_{i=0}^{N-1} |P_i \cap L_i|}{\sum_{i=0}^{N-1} |P_i \cap L_i| + \sum_{i=0}^{N-1} |P_i - L_i|} \tag{4}$$

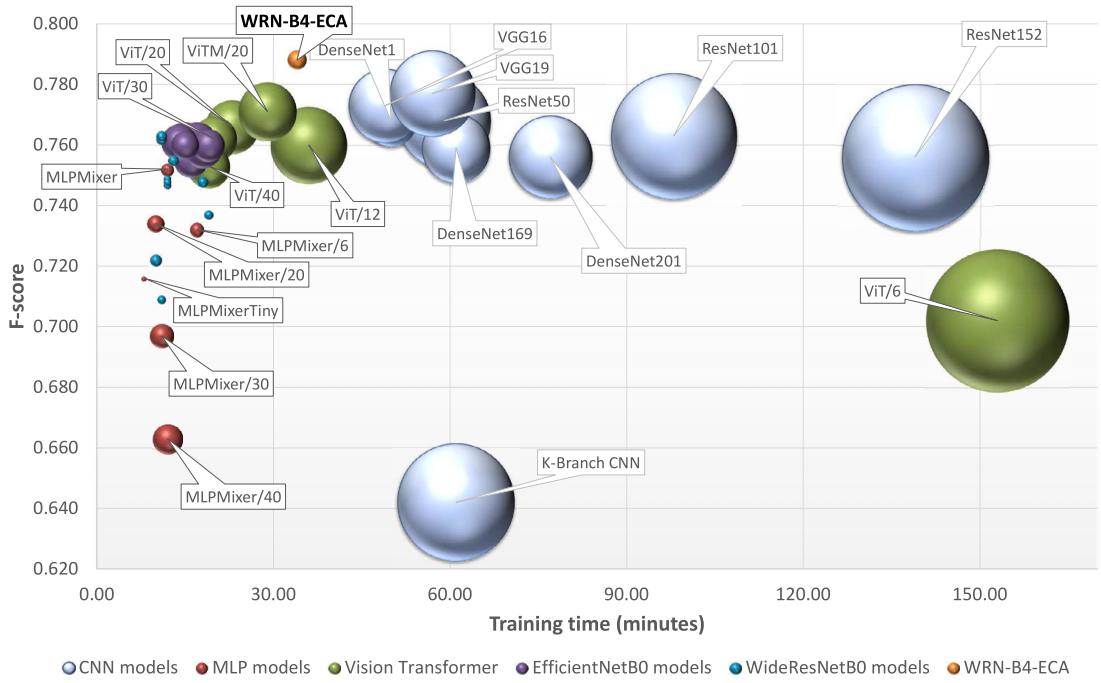

Fig. 6. Performance vs training time trade-off for the different models evaluated. All models of Table 3 are included, plus our best performing model of Table 5: WRN-B4-ECA. The size of the bubbles is proportional to the size of the model. Training time is estimated on a cluster of 20 T K40 GPUs.

Micro Recall (recall averaged over all the label pairs) is defined as:

$$R_{micro} = \frac{TP}{TP + FN} = \frac{\sum_{i=0}^{N-1} |P_i \cap L_i|}{\sum_{i=0}^{N-1} |P_i \cap L_i| + \sum_{i=0}^{N-1} |L_i - P_i|} \tag{5}$$

Micro F Measure by label is the harmonic mean between Micro Precision and Micro Recall.

$$F_{micro} = 2 \cdot \frac{TP}{2 \cdot TP + FP + FN} = \\ = 2 \cdot \frac{\sum_{i=0}^{N-1} |P_i \cap L_i|}{2 \cdot \sum_{i=0}^{N-1} |P_i \cap L_i| + \sum_{i=0}^{N-1} |L_i - P_i| + \sum_{i=0}^{N-1} |P_i - L_i|} \tag{6}$$

We use in this benchmark the Micro set of metrics (Eq. (4), (5), (6)) for optimization, hyperparameter tuning and reporting the results on the test set in Table 3.

4.3. Benchmark implementation

We provide a framework, built upon and extending the implementation in Sumbul et al. (2020), for efficient distributed training in TensorFlow API2 (Abadi et al., 2016). We use Horovod (Sergeev and Balso, 2018), which is a high-level API that sits on top of TensorFlow for training on multiple nodes and GPUs. In Sergeev and Balso (2018), it is argued that Horovod optimally utilizes the available network to take full advantage of hardware resources, therefore one could gain better performance than using a pure Distributed TensorFlow implementation. Hence, we publish on our github repository the distributed implementation of the deep neural networks pipelines created for this work, along with the pretrained weights to facilitate uptake of novel transfer learning applications based on Sentinel-2 data.

We conducted our experiments on Aris High Performance Computing (HPC) infrastructure, and used 10 nodes with 2 GPU-NVIDIA Tesla K40 each. In our experiments, we focus on two directions: speed and classification accuracy. We examine how fast we can train quality classifiers, how fast we can classify new samples at inference, and finally, how well our classifiers perform. The metrics systematically recorded are (i) training time (hours and minutes), (ii) inference rate (images per second), (iii) Precision (Eq. (4)), (iv) Recall (Eq. (5)), (v) Accuracy, and (vi) F-Score (Eq. (6)).

4.4. Results

We report the benchmark results in Table 3 for the eight CNN, two MLP, six ViT, eight EfficientNet-based and ten WRN-based models. In addition, we show in Fig. 6 the models' performance as a function of the F-score metric (Eq. (6)) to capture the classification accuracy, the training time to capture the efficiency of the network, and the total number of trainable parameters. For completeness, we include in Table 4 the metrics from the original BigEarthNet paper (Sumbul et al., 2020) for the CNN architectures tested.

According to Table 3, the two best performing models are EfficientNetB0-ECA and WRNB0-ECA. Together with a vanilla EfficientNetB0 model as it is available from Keras, we scale them from B0 to B7 using the compound scaling method (Section 3.4). The main difference between our EfficientNet-SE implementation and the EfficientNet-Keras one is the use of the reduction ratio (r) within the Squeeze and Excitation module. We use the reduction ratio suggested by the SE paper authors in Hu et al. (2018), instead of the one suggested by the original EfficientNet paper authors (Tan and Le, 2019). We present the model scaling results in Table 5.

5. Discussion

5.1. Trade-offs in the benchmark

Training times vary considerably for the traditional CNN family of models, ranging from ~50 min to over two hours, while F-score varies within a 2% interval. Overall VGG prevails. VGG19 achieves the highest accuracy, trained in 57 min, while VGG16 is the fastest to train (49 min), even though it has more parameters than some of its CNN counterparts, e.g. DenseNet169. At inference though, the fewer the overall parameters, the higher the processed images per second rate, and here DenseNet121 performs best. The CNN model results of Table 3 match well with the metrics reported in Sumbul et al. (2020) (Table 4). Furthermore, we train a ResNet50 model on only 1 GPU-NVIDIA Tesla K40, in order to appreciate the improvement in training time with our distributed learning implementation (Section 4.3). Rows 1 and 2 in Table 3 highlight the 13.5× speedup with our implementation. Moreover,

Table 3
Results of the benchmark, conducted with the distributed learning framework, for the eight CNN, six MLP, six Vision Transformer, eight EfficientNet-based, one K-Branch, and ten WRN-based models. ResNet50^{1GPU} metrics correspond to the ResNet50 model when trained in one GPU only. For each model family, we highlight in bold the best metric achieved, concerning F-score, training time, and inference rate. EfficientNetB0-ECA and WRNB0-EC, in bold, are the two best performing models in the benchmark, which we select to scale through compound scaling (Section 3.4).

Model	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)	Training time (h.min)	Inference rate (imgs/s)	Model size
ResNet50^{1GPU}	61.8	78.1	74.8	**76.4**	13.22	345	23,648,595
ResNet50	62.4	78.8	75.0	76.8	0.59	351	23,648,595
ResNet50-GHOST	58.4	75.2	72.4	73.8	1.04	403	11,930,387
ResNet101	61.7	78.8	74.0	76.3	1.38	268	42,719,059
ResNet152	60.8	76.2	75.0	75.6	2.19	203	58,431,827
DenseNet121	62.3	79.2	74.5	76.8	0.50	**407**	**7,078,931**
DenseNet169	61.1	77.7	74.1	75.9	1.01	370	12,696,467
DenseNet201	60.8	78.1	73.3	75.6	1.17	325	18,380,435
VGG16	63	81.4	73.6	77.3	**0.49**	242	14,728,467
VGG19	63.6	81.7	74.2	**77.7**	0.57	218	20,038,163
K-Branch	47.3	63.4	65.1	64.2	1.01	510	36,979,027
MLPMixer/6	57.8	76.1	70.6	73.2	0.17	654	468,083
MLPMixer	60.3	79.1	71.8	**75.2**	0.12	807	446,723
MLPMixer/20	58	78.1	69.4	73.4	0.10	780	740,355
MLPMixer/30	53.5	77.1	63.6	69.7	0.11	845	1,369,715
MLPMixer/40	49.6	75.1	59.4	66.3	0.12	823	2,261,991
MLPMixerTiny	55.7	77.5	66.5	71.6	**0.08**	**911**	**40,863**
ViT/6	54.1	76.8	64.6	70.2	2.33	241	55,237,395
ViT/12	61.3	80.7	71.9	76	0.36	668	15,984,915
ViT/20	62.1	80.5	73.1	76.6	0.23	**720**	7,760,147
ViT/30	61.6	80.1	72.7	76.2	0.20	704	5,458,707
ViT/40	60.4	80.1	71	75.3	**0.19**	691	**4,989,203**
ViTM/20	62.7	80.6	73.8	**77.1**	0.29	586	9,286,419
EfficientNetB0-SE	61.4	79.6	72.8	76.1	0.15	640	4,411,251
EfficientNetB0-SE-GHOST	60.7	80.2	71.4	75.5	0.16	602	3,053,251
EfficientNetB0-CBAM	61.5	79.9	72.7	76.1	0.17	501	4,412,819
EfficientNetB0-CBAM-GHOST	61.0	80.5	71.7	75.8	0.18	471	3,054,819
EfficientNetB0-COORD	61.2	79.3	72.8	75.9	0.18	604	4,191,967
EfficientNetB0-COORD-GHOST	61.3	80.0	72.4	76.0	0.19	509	2,833,967
EfficientNetB0-ECA	61.4	79.6	72.9	**76.1**	**0.14**	**651**	**3,461,663**
EfficientNetB0-ECA-GHOST	61.4	80.4	72.1	76.0	0.15	611	2,103,663
WRNB0	56.5	80.0	65.9	72.2	0.10	807	306,803
WRNB0-GHOST	54.9	79.5	64.0	70.9	0.11	845	157,619
WRNB0-SE	61.5	81.1	71.8	76.2	0.11	751	309,729
WRNB0-SE-GHOST	59.6	80.8	69.5	74.7	0.12	808	160,545
WRNB0-CBAM	60.6	80.2	71.3	75.5	0.13	639	310,023
WRNB0-CBAM-GHOST	59.9	80.0	70.5	74.9	0.12	670	160,839
WRNB0-COORD	59.8	80.9	69.6	74.8	0.18	588	312,747
WRNB0-COORD-GHOST	58.3	80.4	68.0	73.7	0.19	655	163,563
WRNB0-ECA	61.7	81.5	71.8	**76.3**	**0.11**	**823**	**306,817**
WRNB0-ECA-GHOST	59.7	80.7	69.7	74.8	0.12	851	157,633

Table 4
Model metrics reported by the original BigEarthNet paper (Sumbul et al., 2020). Metrics match relatively well with our implementations presented in Table 3. Our VGG based model results are slightly better and our ResNet-based model results are slightly worse.

Model	Precision	Recall	F-score
ResNet50	81.39	77.44	**77.11**
ResNet101	80.18	77.45	76.49
ResNet152	81.72	76.24	76.53
VGG16	81.05	75.85	76.01
VGG19	79.87	76.71	75.96

we evaluate the impact, the Ghost module has on the ResNet50 model. We notice a 50% decrease of the model parameters and therefore an improvement inference rate, accompanied by an almost 8% increase in the training time, as well as a 3% drop of the F-score, compared to the ResNet50 model. Our results for the ResNet50-GHOST model, match well with the results reported by Han et al. (2020).

The MLPmixer-based models are very efficient in training and inference, but are slightly below average in classification accuracy. In practice, these models can achieve performance similar to the more heavyweight CNN models, with significantly less parameters, from as little as 40 thousand parameters. Large CNN models, such as ResNet152 and DenseNet201, achieve almost the same accuracy as MLPmixer, but use around 58 and 18 million parameters respectively. Such a large model capacity has negative effects to both training time and memory usage. MLPMixer presents a good balance between training time and overall accuracy, with just under half a million trainable parameters.

On the other hand, our MLPMixerTiny model has the fewest trainable parameters, for example, three orders of magnitude less compared with ResNet101, and is trained for 30 epochs in just eight minutes, which is the fastest time in the benchmark. This is a 6× improvement with respect to the fastest traditional CNN model, VGG16, and a 17× improvement with respect to the slowest to train CNN model, ResNet152. This comes at the expense of accuracy though: with 71.6% F-score, MLPMixerTiny is at the bottom of the list, hence in this case there is a trade-off between super-fast training time versus a drop in classification accuracy, as one would expect.

This trade-off does not apply to ViT, EfficientNets and WRNs models, for different reasons. Interestingly, ViT/6 produces the lowest accuracy and is also the most difficult to train. It experiences the worse overall performance (accuracy and training time) in the study. Examining the effect of the patch size, we show that very small values have a negative effect on the performance of the model. Patch size of 20 produces the best results with ViT/20 achieving 77.1% F-Score, while retaining the training time under half an hour. This is matching the

Table 5
Scaling and training: (i) our best performing model: WRNB0-ECA in this table, (ii) our EfficientNetB0-ECA (here EfNetB0-ECA) according to Table 3, and (iii) a vanilla EfficientNetB0-Keras (here EfNetB0-Keras) architecture as available from Keras.

Model size	Model	Precision (%)	Recall (%)	F-score (%)	Training time (h.mm)
WRN-B0-ECA	306,817	81.5	71.8	76.3	0.11
EfNetB0-ECA	3,461,663	79.6	72.9	76.1	0.14
EfNetB0-Keras	4,075,940	75.6	70.9	73.1	0.34
WRN-B1-ECA	373,381	81.8	72.5	76.9	0.12
EfNetB1-ECA	5,511,623	81.1	73.2	77.0	0.19
EfNetB1-Keras	6,601,608	75.9	71.4	73.6	0.49
WRN-B2-ECA	433,209	82.2	73.1	77.4	0.19
EfNetB2-ECA	6,503,649	81.3	73.8	77.3	0.27
EfNetB2-Keras	7,797,370	75.0	70.6	72.7	0.51
WRN-B3-ECA	590,333	82.4	74.4	78.2	0.23
EfNetB3-ECA	8,981,821	81.7	74.0	77.7	0.36
EfNetB3-Keras	10,815,272	76.4	71.7	74.0	1.03
WRN-B4-ECA	985,961	**82.4**	75.5	78.8	**0.34**
EfNetB4-ECA	14,630,489	81.7	73.7	77.5	1.00
EfNetB4-Keras	17,710,928	73.9	72.7	73.3	1.22
WRN-B5-ECA	5,166,299	82.0	**76.1**	**79.0**	2.46
EfNetB5-ECA	23,454,139	80.6	73.9	77.1	1.37
EfNetB5-Keras	28,555,496	78.4	74.2	76.2	1.55
WRN-B6-ECA	7,281,895	82.1	75.1	78.5	3.50
EfNetB6-ECA	33,591,965	81.3	74.0	77.4	2.17
EfNetB6-Keras	41,007,480	75.0	72.2	73.6	2.26
WRN-B7-ECA	14,068,791	79.6	73.5	76.4	7.50
EfNetB7-ECA	52,340,949	81.6	71.4	76.1	4.09
EfNetB7-Keras	64,150,392	78.9	73.5	76.1	4.45

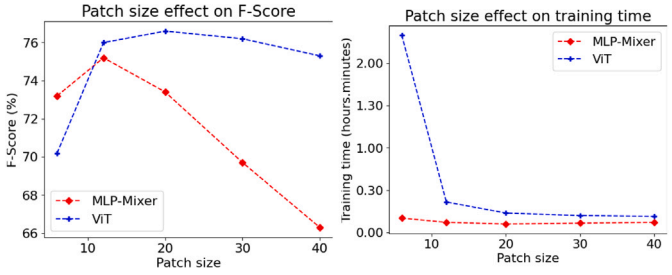

Fig. 7. Ablation on the effect of patch size on model's performance in regards to F-Score and training time for both MLPMixer and Vision Transformer.

accuracy of the best CNNs, but the model is trained faster. A similar effect is observed for the MLP-Mixer. Our base version with patch size equal to 12 produces the best F-Score while maintaining a good training time. Increasing the patch size too much significantly hurts the performance of MLP-Mixer with no considerable gain in training speed. The effect of patch size is summarized in Fig. 7 for both architectures vis-à-vis F-Score and training time.

EfficientNets and even more WRNs model families, exhibit the best overall performance considering the trade-offs between training time, F-score, and inference rate. This happens even for the non-scaled, simpler B0 models. Both EfficientNet and WRN models share four common characteristics: (i) they achieve 4× to 10× faster training times compared to CNNs, (ii) the inclusion of the Ghost module deteriorates their performance mainly in terms of classification accuracy, in contrast to the author claims, while reducing the models' size by ∼35% and ∼50% for EfficientNets and WRNs respectively. (iii) even though including the Ghost module significantly reduces the models' size, those models require a fraction of additional time to train compared to the models without the Ghost module. This is because the Ghost module implementation is more suitable for ARM/CPUs and is not GPU-friendly due to the Depthwise Convolution operations, and therefore is more appropriate to deploy in mobile devices and other devices with limited resources. For these reasons, we believe that the Ghost versions of our models are a promising candidate for Inference-at-the-Edge use cases. Finally, (iv) the ECA attention module provides the best overall results for F-score and training time. WRNs in particular perform best (Fig. 6 and Table 3), considering the fact that they have one order of magnitude fewer parameters with respect to EfficientNets. This does not come at the expense of classification accuracy and on the contrary, it is directly translated to better inference times. Finally, it is noteworthy that classification accuracy variance is negligible for the variants of EfficientNet based models (Fig. 6), while this does not apply to WRN variants. The attention mechanism in WRN affects their performance significantly.

5.2. The new SOTA for BigEarthNet

EfficientNetB0-ECA and WRNB0-ECA models (Table 3) are scaled through compound scaling and the results are reported in Table 5. According to Table 5, we select our top model to be WRN-B4-ECA with an F-score of 78.8% trained in 34 min and processing 381.0 images/sec at inference. According to our literature review, this is the new SOTA for the BigEarthNet dataset. Although WRN-B5-ECA reaches an F-score of 79.0%, the 0.2% gain is not enough to justify the 5× training time and model parameters. In fact, we observe that for WRN-B6-ECA and WRN-B7-ECA models, F-score drops, hinting at overfitting. Additional training data would be needed to estimate the weights for these larger models.

The best model in terms of classification accuracy, trained in the original BigEarthNet paper (Sumbul et al., 2020) is ResNet50 with approximately 23 million parameters. In that work, the authors achieved an image classification F-score of 77.11% (Table 4), while with our setup we reached 76.8% and trained it in almost one hour (Table 3). On the other hand, our lighter WRN-B4-ECA model with one million parameters only, manages an F-score of 78.8%, and is trained on the same data splits in 34 min, which constitutes a significant improvement. This is visually highlighted in Fig. 6. We further investigate this improvement by looking into the metrics for each one of the 19 classes that exist in the dataset. These are shown in Table 6 for the baseline ResNet50 model, vis-à-vis our WRN-B4-ECA model. For every single class, our model outperforms the baseline. For some classes

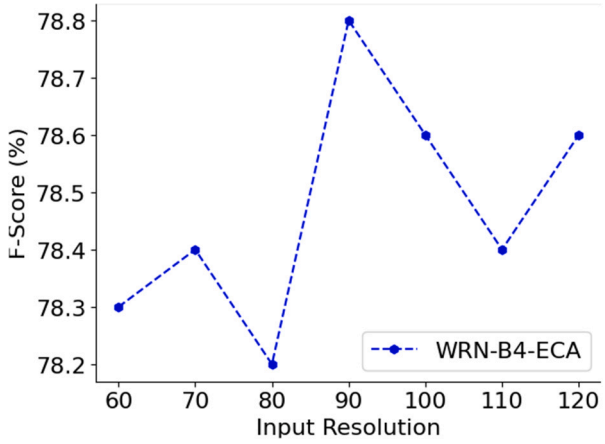

Fig. 8. The effect of various input resolutions on the performance of our WRN-B4-ECA model.

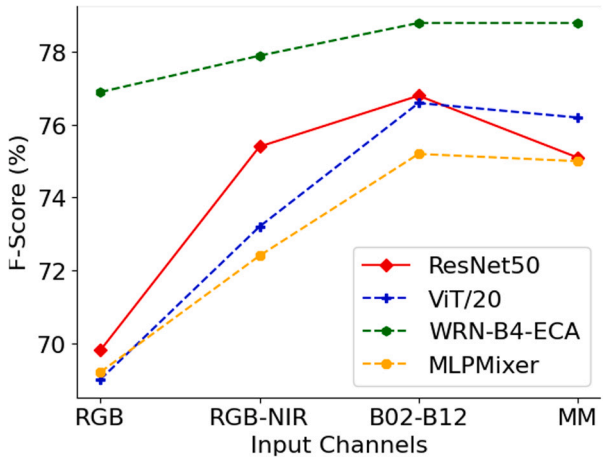

Fig. 9. Ablation on the effect of input channels. X-axis represents the input setting. RGB refers to models trained solely with the RGB bands as input, RGB-NIR corresponds to models trained with RGB and the NIR channel, B02–B12 is our normal setup and finally MM combines both B02–B12 channels of Sentinel-2 and VV-VH channels of Sentinel-1.

the difference is remarkable, for example, class 'permanent crops' is better resolved with an F-score jump from 52% to 65.6%. Similar improvement jumps are noted for 'transitional woodland-shrub' and 'coastal water' classes. Overall, the averaged metrics in the last line of Table 6, which correspond to Macro F-score instead of the Micro F-score measures in Eq. (6), show an increase by almost 4.47% when using our lighter model.

5.3. Examination of the effect of input resolution on our SOTA model

Following the designation of our WRN-B4-ECA model as the new SOTA for BigEarthNet, we investigate whether we could benefit in terms of performance by altering the input image resolution. The default input resolution used by our WRN-B4-ECA model is 90×90, while it ranges during the course of our scaling experiments from 60×60, up to 120×120. The impact of the different input image resolutions is summarized in Fig. 8. Our investigation demonstrates that the default input resolution used by the WRN-B4-ECA model is ideal and results in the highest performance among all the other variants.

5.4. Examination of the effect of input channels

Motivated by the introduction of the SAR modality (Sentinel-1) in the second version of BigEarthNet dataset (Sumbul et al., 2021b), we investigate whether the extra information provided by the SAR images and the different multispectral channels of Sentinel-2 is actually beneficial for the task at hand. Our investigation revolves around four diverse architectures, i.e. ViT, ResNet50, MLPMixer and our best performing model WRN-B4-ECA. We define the following settings for our experiments. First, we experiment with inputs consisting solely of the RGB channels. We then augment our input by introducing the NIR band and train with 4-channel inputs. Since we have already conducted the experiments with all multispectral (B02–B12) channels, we proceed with the introduction of the Sentinel-1 modality resulting in a 12-channel input. The results of this ablation are summarized in Fig. 9. From our experiments, it is clear that the multispectral information is very beneficial for the task of land use land cover classification. This finding, emphasizes our initial intuition that the restriction on the input channels (RGB) induced by models pretrained on ImageNet results in loss of information.

On the other hand, Sentinel-1 data do not seem to contribute much for this task. This could be attributed to the nature of the dataset. BigEarthNet is focused on frames with minimum cloud coverage. SAR data could prove to be really helpful in scenarios of high cloud coverage, since the radar microwave frequencies can penetrate clouds providing some backscatter information, as opposed to optical data that are completely obscured by clouds.

Finally, these experiments emphasize the superiority of our proposed WRN-B4-ECA model, as it consistently outperforms the rest of the models and maintains its performance even after the introduction of the Sentinel-1 images contrary to the rest of the models in this experiment.

Table 6
BigEarthNet class-based F-scores (%) for our best performing WRN-B4-ECA model according to Table 5, and a ResNet50 baseline CNN model.

BigEarthNet class	BigEarthNet ResNet50	Our WRN-B4-ECA
Urban fabric	74.84	75.17
Industrial or commercial units	48.55	49.14
Arable land	83.85	86.25
Permanent crops	51.91	65.49
Pastures	72.38	76.27
Complex cultivation patterns	66.03	70.27
Land principally occupied by agriculture, with significant areas of natural vegetation	60.94	65.89
Agro-forestry areas	70.49	77.89
Broad-leaved forest	74.05	80.48
Coniferous forest	85.41	86.97
Mixed forest	79.44	81.24
Natural grassland and sparsely vegetated areas	47.55	51.28
Moors, heathland and sclerophyllous vegetation	59.41	62.54
Transitional woodland-shrub	53.47	68.1
Beaches, dunes, sands	61.46	66.67
Inland wetlands	60.64	58.47
Coastal wetlands	47.71	63.16
Inland waters	83.69	86.06
Marine waters	97.53	98.57
Average	**67.33**	**71.8**

5.5. Model explainability

We use Gradient-weighted Class Activation Mapping (Grad-CAM), as in Selvaraju et al. (2017), in order to understand some of the

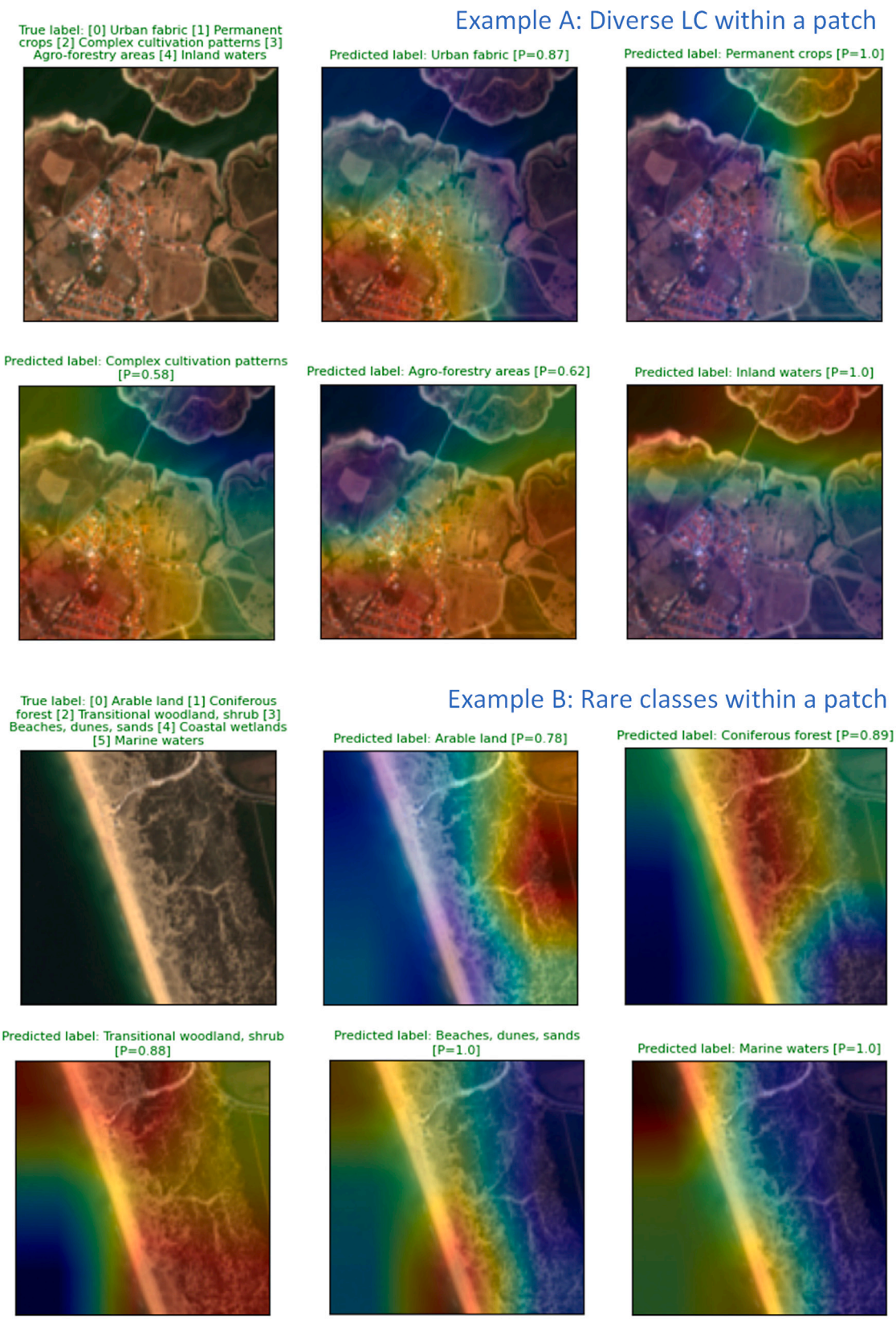

Fig. 10. Examples of two challenging image patches, correctly classified. The first patch in Examples A & B is the original Sentinel-2 image patch with the different LULC classes contained. The other patches are the output of Grad-CAM (Selvaraju et al., 2017) that we adopt to interpret which parts of the image were used by our network for deciding on each specific True Positive LULC class. P stands for the probability a specific LULC is contained in the patch.

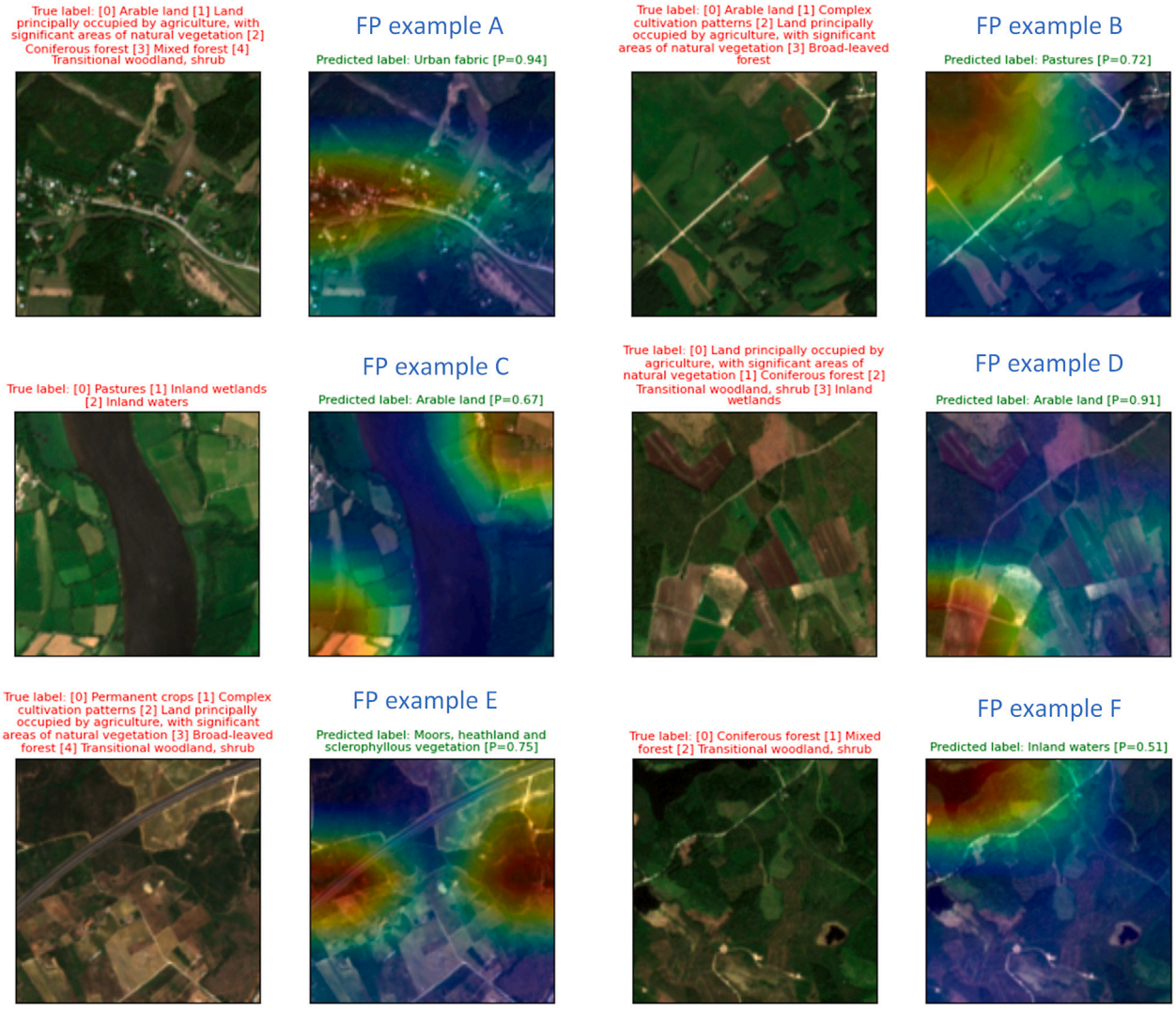

Fig. 11. Examples of image pairs with False Positive LULC scene classification. For each pair we show on the left the original Sentinel-2 image patch with all the LULC classes contained, and we show on the right the Grad-CAM (Selvaraju et al., 2017) output for a False Positive class. Red areas correspond to the part of the image patch used to make the False Positive prediction. P stands for the probability a specific LULC is contained in the patch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

image classification accuracy discrepancies observed in the benchmark. Grad-CAM produces 'interpretable' explanations for the classification decisions of our resulting model. It exploits the gradients of logits for the different classes of the final convolutional layer to produce a map that highlights the important regions in the image, used for predicting a specific class.

In Fig. 10 we have selected two challenging Sentinel-2 image patches, one in each row, that contain several LULC classes. We also show the Grad-CAM output for the True Positive classes predicted, with the associated probability P that a specific LULC is contained in the patch. If $P > 0.5$ we assign this class to the patch. On the top of Fig. 10 is an image patch with urban, agricultural, vegetated and marine areas, and all different classes are correctly resolved, while the classification decision seems to focus on the appropriate parts of the image. On the bottom of Fig. 10 is an image patch with some rare classes, e.g. *Beaches, dunes, sands* and *Transitional woodland, shrub*, which are predicted with high probabilities, and again focusing on the correct part of each image.

In Fig. 11 we provide False Positive samples and the corresponding Grad-CAM output. Investigating these Grad-CAM outputs and relying on the visual content of the visible spectral channels only, it can be argued that it is challenging for the human eye to reject the predicted False Positive LULC classes as well. For example, the *Urban fabric* predicted class in Fig. 11, indeed focuses on settlements or individual buildings that exist in the original image patch. Similarly, the *Pastures* predicted class cannot be easily dismissed, especially when not all image spectral content is visualized. FP class *Arable land* is attributed to patches that according to Grad-CAM output indeed focus on agricultural-like areas. In this case, the higher level category is correct, i.e. agricultural areas, while the more detailed LULC class in the taxonomy is not correct, i.e. *Arable land* is predicted instead of *Land principally occupied by agriculture, with significant areas of natural vegetation*. The last sample in Fig. 11 contains an *Inland waters* FP class prediction, and the network focuses on the upper left part of the patch which indeed could be attributed to a water body. Overall, it could be argued that the FP predictions are within the error margin even of an experienced remote sensing photo-interpreter.

Finally, in Fig. 12 we show an example of an image patch that contains seven different LULC classes, of which five are correctly predicted and two are False Negatives. Grad-CAM outputs for the TP classes again focus visually on the correct parts of the patch. The *Inland waters* class especially has a Grad-CAM output that could be potentially used directly for image segmentation. Investigating hundreds Grad-CAM samples in our test dataset, the same can be inferred for several other LULC predictions. The FN LULC classes are hard to predict,

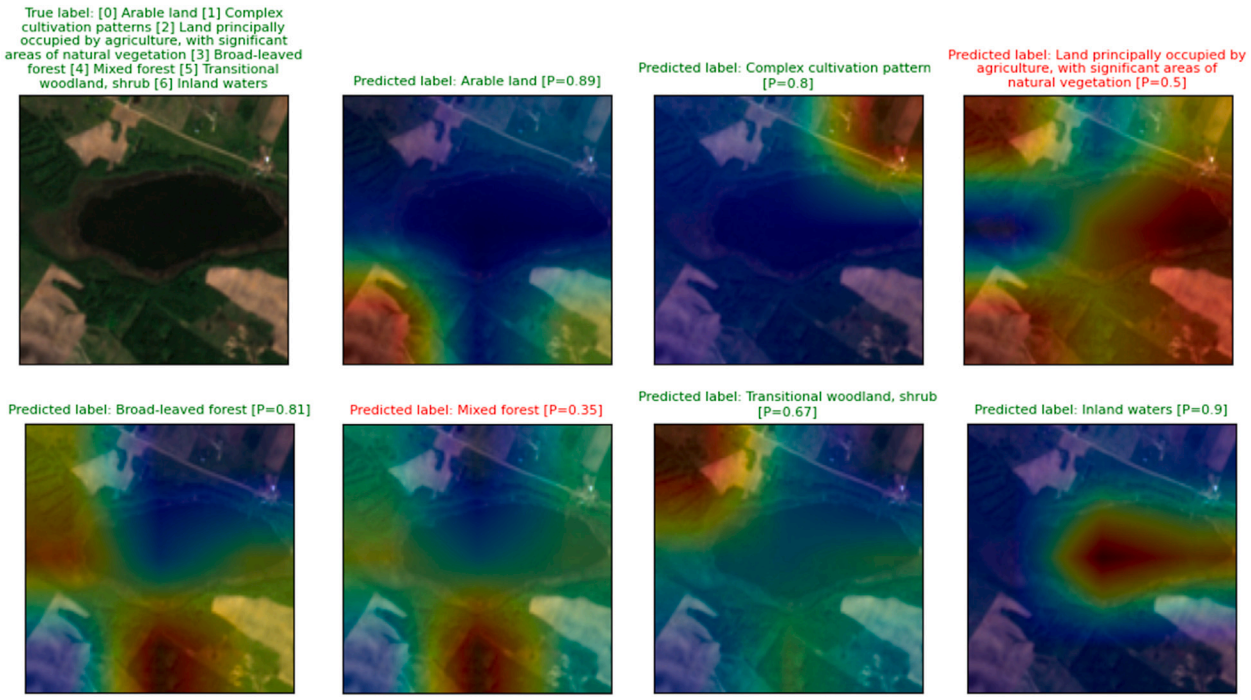

Fig. 12. Example of an image patch with True Positive and False Negative LULC scene classification. The top-right image is the original Sentinel-2 image patch with all the LULC classes contained. The other images are the output of Grad-CAM (Selvaraju et al., 2017) for interpreting which parts of the original patch were used by our network for deciding on each specific True Positive (green font) or False Negative (red font) LULC class.P stands for the probability a specific LULC is contained in the patch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

even for an expert in satellite image photointerpretation. Indicatively, while Grad-CAM focuses on densely vegetated areas in Fig. 12, our network correctly predicts the *Broad-leaved forest* class but not the *Mixed forest* class. These two classes, however, are almost indistinguishable considering their spectral signatures. Therefore it could be worth investigating a new taxonomy by merging LULC classes, for which the spectral content and spatial patterns are so similar that even deep neural networks cannot confidently resolve.

5.6. Transfer learning

BigEarthNet revolves around a specific RS task, LULC scene classification with a specific set of classes. In other RS tasks that use Sentinel-2 imagery, researchers are developing new training datasets on an ad-hoc basis. Creating an independent, unique, labeled dataset for each RS problem is not feasible, given the plethora of tasks in the RS domain. The paradigm in the Computer Vision community is different. The publicly available models trained on ImageNet natural images have been successfully and extensively used in various transfer learning applications. Motivated by the impact of this approach, we believe that a similar logic should be adopted by the remote sensing community. However, transferring knowledge from such a different domain is not optimal. Having this in mind, we provide a Sentinel-2 domain-relevant pre-trained model zoo, which can be subsequently used to the fullest for different transfer learning RS applications.

We put this argument to the test, with an extensive study on the performance of our models pretrained on BigEarthNet for new RS tasks. We split our investigation in two experiments on two different datasets: EUROSAT (Helber et al., 2019) and SEN12MS (Schmitt and Wu, 2021), introduced in Section 2.2. EUROSAT is a Sentinel-2 based dataset addressing the task of land cover classification. Given that there is no standard dataset split, we divide EUROSAT in three sets for training, validation and testing with a ratio of 80/10/10. SEN12MS is the second Sentinel based dataset we use for this study, which contains both Sentinel-1 and Sentinel-2 modalities. With SEN12MS, we evaluate our models on a multi-label scene classification problem, and it is a more challenging dataset compared to EUROSAT. The current state of the art reported in Schmitt and Wu (2021) achieves at most 69.9% F-Score, when using only the Sentinel-2 modality as input to a ResNet50 and 72.0% with a DenseNet121 when combining both Sentinel-1 and Sentinel-2. Furthermore, the authors notice the importance of the multispectral information, observing a drop in performance when ignoring it. On the other hand, the authors of Helber et al. (2019) achieve very high accuracy on EUROSAT, i.e. 98.56% using solely the RGB channels as input to a ResNet50 model. Moreover, their experiments show that one can achieve very high accuracy >90% while using only one band making the multispectral information redundant. A reason for this could be the choice of the classes in EUROSAT, consisting mainly of high level categories that could be discriminated by the shapes, texture and color e.g. Sea&Lake versus Highway. On the contrary, SEN12MS, similar to BigEarthNet, aims to identify land use and land cover in a more fine-grained manner, containing classes such as Grassland, Cropland and Shrubland.

In our experiments, we investigate (a) the quality of representations learnt on BigEarthNet compared to the respective representations of ImageNet, (b) the benefit of the introduction of pretrained models that can exploit the full multispectral information, and (c) the capacity of these models for good performance in low data regimes. Overall, we observe a consistent improvement induced by the usage of weights learnt on BigEarthNet.

We begin our study by comparing the models pretrained on ImageNet and the same architectures pretrained on BigEarthNet to prove the superiority of in-domain learnt features. We use common architectures with existing ImageNet pretrained weights, i.e. ResNet50 and DenseNet121. Additionally, we include our top performing model WRN-B4-ECA to examine its performance on different datasets. Since our derived model is introduced in this work there are no weights pretrained on ImageNet. Given that ImageNet contains only RGB images, we include in our study the respective setting pretrained on BigEarthNet. For our first experiment then, summarized in Fig. 13, we

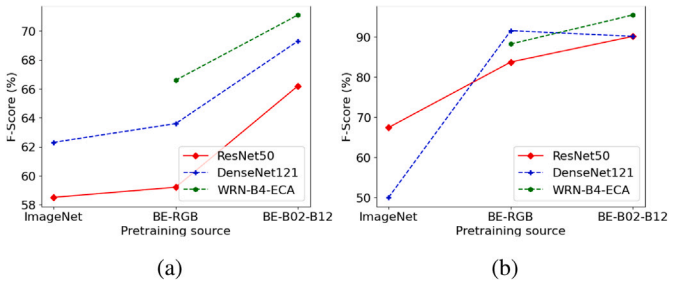

Fig. 13. Finetuning of ResNet50, DenseNet121 and WRN-B4-ECA under different pretraining schemes. We present the results of the evaluation on SEN12MS and EUROSAT in (a) and (b) respectively.

finetune our pretrained models on SEN12MS and EUROSAT examining three pretraining schemes: pretraining on ImageNet, pretraining on BigEarthNet using solely the RGB channels as input, and pretraining on BigEarthNet using the full multispectral information as done in Table 3. We allow all layers to be trainable in the finetuning phase. The superiority of the models pretrained on BigEarthNet is a fact for all experiments (Fig. 13), even when the models were trained solely with RGB inputs. All models benefit by the transition from ImageNet to BigEarthNet pretraining scheme. WRN-B4-ECA seems to consistently perform the best, achieving the best results when fed with multispectral inputs. This information could not be harnessed by models pretrained on ImageNet without any architecture modification. Nevertheless, these results are achieved after training on full, curated datasets. To evaluate the impact of in-domain transfer learning for RS scenarios where creating a new large annotated dataset is not feasible, we have to investigate the performance of our models in different data-regimes.

Hence, in our second experiment, we follow this intuition and investigate the behavior of our models in lower data regimes. For this examination we focus on ResNet50 and WRN-B4-ECA trained on variants of SEN12MS with the multispectral channels as input. SEN12MS's volume allows us to examine a wider range of dataset sizes. We attempt to learn the classification task defined in SEN12MS using the 1%, 10%, 20%, 50% and 100% of the training dataset and observe the difference in performance when compared to random initialization. In both initialization settings, we train the whole network. The results of this experiment are shown in 14. As expected, pretraining on large curated datasets such as BigEarthNet can significantly improve performance for related data. Both models pretrained on BigEarthNet perform consistently better than their randomly initialized counterparts. When training with very small training datasets, the performance boost induced by pretraining is very high for both architectures. Impressively, WRN-B4-ECA is able to achieve state-of-the art performance (71.1% F-Score) with just 10% of the dataset. Naturally, as the dataset size increases the need for pretrained weights is limited and the difference between random initialization and pretraining diminishes.

These findings highlight the fact that the pretrained model-zoo can be beneficial for RS tasks with small labeled sets, as well as for research labs with low computational resources. Furthermore, by providing a large enough and reproducible benchmark, the evaluation of future methods becomes more transparent. Finally, the usage of pretrained models as well as a common reference benchmark alleviates the need for repeating expensive experiments leading to reduced carbon footprint.

6. Conclusion

We address the multi-label, multi-class LULC single Sentinel-2 image classification problem and benchmark several popular deep learning architectures and more sophisticated models, such as ViT. We develop and use a distributed learning implementation, and create a model

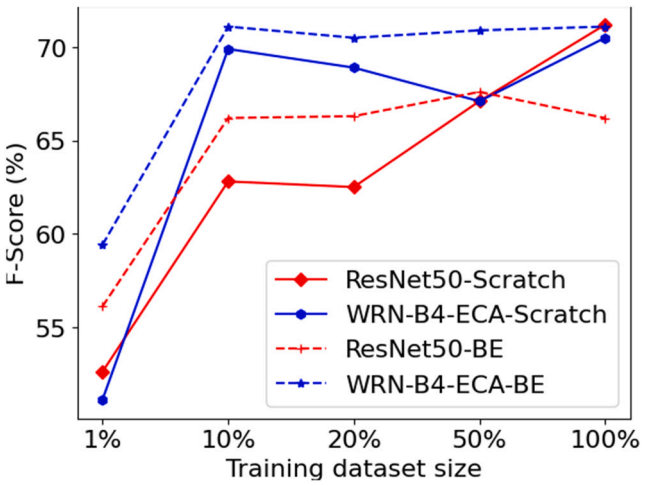

Fig. 14. Investigation of transfer learning on low data regimes on SEN12MS. We indicate the ResNet50 with red and the WRN-B4-ECA with blue. The dashed lines show the performance of the models pretrained on BigEarthNet(BE) while the solid lines the performance of models trained from scratch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

zoo of 62 trained models, which we make publicly available in order to boost research in diverse tasks that exploit multispectral imagery. Considering the challenges in training on big satellite datasets, we seek to optimize model performance jointly in terms of training time, inference rate and classification accuracy. We find that through compound scaling of lightweight Wide Residual Networks we achieve the best overall performance. Our lightweight Wide Residual Network model with an Efficient Channel Attention mechanism and scaled by adapting the EfficientNet compound scaling methodology performs best in our benchmark. It achieves 4.5% higher averaged f-score classification accuracy for all 19 LULC classes or a 2% increase in overall micro f-score, and is trained two times faster compared to a ResNet50 baseline model. Finally, our benchmark reveals that conventional CNN models that perform costly convolutions can be matched by Multilayer Perceptron feedforward artificial neural networks that are more lightweight, much simpler, and faster to train.

Our findings imply that efficient lightweight deep learning models that are fast to train when appropriately scaled for depth, width, and input data resolution can provide comparable and even higher image classification accuracies. This is especially important in remote sensing where the volume of data coming from the Sentinel family but also other satellite platforms is very large and constantly increasing. We believe that our approach for designing light and scalable models can go beyond the specific LULC scene classification problem addressed herein, and could be tested in different application scenarios, e.g. in food security at large scales, and other tasks, such as semantic segmentation and object detection in satellite imagery. However, the potential for transfer learning of the DL models has to be thoroughly investigated, especially considering the spatio-temporal nature of satellite data. As discussed by Sykas et al. (2022), the classification generalization capacity to different years and geographic locations remains a great challenge in RS, and domain adaptation strategies should be adopted to bridge the inherent gaps.

We hope that this extended benchmark will serve as a quick and robust way for the evaluation of new methods, and the produced model zoo will propel deep learning research in currently, untouched, applications of remote sensing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has received funding from the European Union's Horizon2020 research and innovation project DeepCube under grant agreement number 101004188, and by project EO4flood funded by the National Observatory of Athens. In addition, this work was supported by computational time granted from the National Infrastructures for Research and Technology S.A. (GRNET S.A.) in the National HPC facility - ARIS - under project ID pr010006.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X., 2016. TensorFlow: A system for large-scale machine learning. In: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). USENIX Association, Savannah, GA, pp. 265–283, URL https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi.

Aksoy, A.K., Ravanbakhsh, M., Kreuziger, T., Demir, B., 2021. A novel uncertainty-aware collaborative learning method for remote sensing image classification under multi-label noise. CoRR abs/2105.05496 arXiv:2105.05496.

Alhichri, H., Alswayed, A.S., Bazi, Y., Ammour, N., Alajlan, N.A., 2021a. Classification of remote sensing images using EfficientNet-B3 CNN model with attention. IEEE Access 9, 14078–14094. http://dx.doi.org/10.1109/ACCESS.2021.3051085.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C., 2021. Vivit: A video vision transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6836–6846. http://dx.doi.org/10.1109/ICCV48922.2021.00676.

Bai, Y., Gao, C., Singh, S., Koch, M., Adriano, B., Mas, E., Koshimura, S., 2018. A framework of rapid regional tsunami damage recognition from post-event TerraSAR-X imagery using deep neural networks. IEEE Geosci. Remote Sens. Lett. 15 (1), 43–47. http://dx.doi.org/10.1109/LGRS.2017.2772349.

Bazi, Y., Al Rahhal, M.M., Alhichri, H., Alajlan, N., 2019. Simple yet effective fine-tuning of deep CNNs using an auxiliary classification loss for remote sensing scene classification. Remote Sens. 11 (24), http://dx.doi.org/10.3390/rs11242908, URL https://www.mdpi.com/2072-4292/11/24/2908.

Bello, I., Fedus, W., Du, X., Cubuk, E.D., Srinivas, A., Lin, T.-Y., Shlens, J., Zoph, B., 2021. Revisiting ResNets: Improved training and scaling strategies. In: Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (Eds.), Advances in Neural Information Processing Systems. URL https://openreview.net/forum?id=dsmxf7FKiaY.

Ben Hamida, A., Benoit, A., Lambert, P., Ben Amar, C., 2018. 3-D deep learning approach for remote sensing image classification. IEEE Trans. Geosci. Remote Sens. 56 (8), 4420–4434. http://dx.doi.org/10.1109/TGRS.2018.2818945.

Buchhorn, M., Lesiv, M., Tsendbazar, N.-E., Herold, M., Bertels, L., Smets, B., 2020. Copernicus global land cover layers—Collection 2. Remote Sens. 12 (6), http://dx.doi.org/10.3390/rs12061044, URL https://www.mdpi.com/2072-4292/12/6/1044.

Cai, W., Wei, Z., 2020. Remote sensing image classification based on a cross-attention mechanism and graph convolution. IEEE Geosci. Remote Sens. Lett. 19, 1–5. http://dx.doi.org/10.1109/LGRS.2020.3026587.

Cao, R., Fang, L., Lu, T., He, N., 2020. Self-attention-based deep feature fusion for remote sensing scene classification. IEEE Geosci. Remote Sens. Lett. 18 (1), 43–47. http://dx.doi.org/10.1109/LGRS.2020.2968550.

Chaib, S., Liu, H., Gu, Y., Yao, H., 2017. Deep feature fusion for VHR remote sensing scene classification. IEEE Trans. Geosci. Remote Sens. 55 (8), 4775–4784. http://dx.doi.org/10.1109/TGRS.2017.2700322.

Charoenchittang, P., Boonserm, P., Kobayashi, K., Cooharojananone, N., 2021. Airport buildings classification through remote sensing images using EfficientNet. In: 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). pp. 127–130. http://dx.doi.org/10.1109/ECTI-CON51831.2021.9454686.

Chaudhuri, U., Dey, S., Datcu, M., Banerjee, B., Bhattacharya, A., 2021. Inter-band retrieval and classification using the multi-labeled sentinel-2 BigEarthNet archive. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 1. http://dx.doi.org/10.1109/JSTARS.2021.3112209.

Chen, C., Chandra, S., Han, Y., Seo, H., 2022. Deep learning-based thermal image analysis for pavement defect detection and classification considering complex pavement conditions. Remote Sens. 14 (1), http://dx.doi.org/10.3390/rs14010106, URL https://www.mdpi.com/2072-4292/14/1/106.

Chen, J., Huang, H., Peng, J., Zhu, J., Chen, L., Li, W., Sun, B., Li, H., 2020. Convolution neural network architecture learning for remote sensing scene classification. http://dx.doi.org/10.48550/ARXIV.2001.09614, URL https://arxiv.org/abs/2001.09614.

Chen, H., Shi, Z., 2020. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. Remote Sens. 12 (10), http://dx.doi.org/10.3390/rs12101662, URL https://www.mdpi.com/2072-4292/12/10/1662.

Copernicus, 2018. CORINE land cover 2018. URL https://land.copernicus.eu/pan-european/corine-land-cover.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp. 248–255. http://dx.doi.org/10.1109/CVPR.2009.5206848.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Diakogiannis, F.I., Waldner, F., Caccetta, P., Wu, C., 2020. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. ISPRS J. Photogramm. Remote Sens. 162, 94–114. http://dx.doi.org/10.1016/j.isprsjprs.2020.01.013, URL https://www.sciencedirect.com/science/article/pii/S0924271620300149.

Ding, L., Tang, H., Bruzzone, L., 2020. LANet: Local attention embedding to improve the semantic segmentation of remote sensing images. IEEE Trans. Geosci. Remote Sens. 59 (1), 426–435. http://dx.doi.org/10.1109/TGRS.2020.2994150.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. CoRR abs/2010.11929 arXiv:2010.11929.

Du, S.S., Zhai, X., Poczos, B., Singh, A., 2018. Gradient descent provably optimizes over-parameterized neural networks. In: International Conference on Learning Representations. http://dx.doi.org/10.48550/ARXIV.1810.02054.

Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., Dean, J., 2019. A guide to deep learning in healthcare. Nat. Med. 25 (1), 24–29. http://dx.doi.org/10.1038/s41591-018-0316-z.

Fan, R., Feng, R., Wang, L., Yan, J., Zhang, X., 2020. Semi-MCNN: A semisupervised multi-CNN ensemble learning method for urban land cover classification using submeter HRRS images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 13, 4973–4987. http://dx.doi.org/10.1109/JSTARS.2020.3019410.

Ghaffarian, S., Valente, J., Van Der Voort, M., Tekinerdogan, B., 2021. Effect of attention mechanism in deep learning-based remote sensing image processing: A systematic literature review. Remote Sens. 13 (15), 2965. http://dx.doi.org/10.3390/rs13152965.

Gómez, P., Meoni, G., 2021. MSMatch: Semisupervised multispectral scene classification with few labels. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14, 11643–11654. http://dx.doi.org/10.1109/JSTARS.2021.3126082.

Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., Xu, C., 2020. GhostNet: More features from cheap operations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 1577–1586. http://dx.doi.org/10.1109/CVPR42600.2020.00165.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 770–778. http://dx.doi.org/10.1109/CVPR.2016.90.

Helber, P., Bischke, B., Dengel, A., Borth, D., 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 12 (7), 2217–2226. http://dx.doi.org/10.1109/JSTARS.2019.2918242.

Hong, D., Hu, J., Yao, J., Chanussot, J., Zhu, X., 2021. Multimodal remote sensing benchmark datasets for land cover classification with a shared and specific feature learning model. ISPRS J. Photogramm. Remote Sens. 178, 68–80. http://dx.doi.org/10.1016/j.isprsjprs.2021.05.011.

Hou, Q., Zhou, D., Feng, J., 2021. Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 13713–13722. http://dx.doi.org/10.1109/CVPR46437.2021.01350.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861.

Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 7132–7141. http://dx.doi.org/10.1109/CVPR.2018.00745.

Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M., Lee, H., Ngiam, J., Le, Q.V., Wu, Y., et al., 2019. Gpipe: Efficient training of giant neural networks using pipeline parallelism. Adv. Neural Inf. Process. Syst. 32.

Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2261–2269. http://dx.doi.org/10.1109/CVPR.2017.243.

Ienco, D., Gaetano, R., Dupaquier, C., Maurel, P., 2017. Land cover classification via multitemporal spatial data by deep recurrent neural networks. IEEE Geosci. Remote Sens. Lett. 14 (10), 1685–1689. http://dx.doi.org/10.1109/LGRS.2017.2728698.

Kakogeorgiou, I., Karantzalos, K., 2021. Evaluating explainable artificial intelligence methods for multi-label deep learning classification tasks in remote sensing. Int. J. Appl. Earth Obs. Geoinf. 103, 102520. http://dx.doi.org/10.1016/j.jag.2021.102520.

Kang, J., Fernandez-Beltran, R., Hong, D., Chanussot, J., Plaza, A., 2021. Graph relation network: Modeling relations between scenes for multilabel remote-sensing image classification and retrieval. IEEE Trans. Geosci. Remote Sens. 59 (5), 4355–4369. http://dx.doi.org/10.1109/TGRS.2020.3016020.

Khatami, R., Mountrakis, G., Stehman, S.V., 2016. A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: General guidelines for practitioners and future research. Remote Sens. Environ. 177, 89–100. http://dx.doi.org/10.1016/j.rse.2016.02.028, URL https://www.sciencedirect.com/science/article/pii/S0034425716300578.

Khurshid, N., Tharani, M., Taj, M., Qureshi, F.Z., 2020. A residual-dyad encoder discriminator network for remote sensing image matching. IEEE Trans. Geosci. Remote Sens. 58 (3), 2001–2014. http://dx.doi.org/10.1109/TGRS.2019.2951820.

Koßmann, D., Wilhelm, T., Fink, G.A., 2021. Towards tackling multi-label imbalances in remote sensing imagery. In: 2020 25th International Conference on Pattern Recognition. ICPR, pp. 5782–5789. http://dx.doi.org/10.1109/ICPR48806.2021.9412588.

Koubarakis, M., Bereta, K., Bilidas, D., Giannousis, K., Ioannidis, T., Pantazi, D.-A., Stamoulis, G., Haridi, S., Vlassov, V., Bruzzone, L., Paris, C., rn Eltoft, T., Krämer, T., Charalabidis, A., Karkaletsis, V., Konstantopoulos, S., Dowling, J., Kakantousis, T., Datcu, M., Dumitru, C.O., Appel, F., Bach, H., Migdall, S., Hughes, N., Arthurs, D., Fleming, A., 2019. From copernicus big data to extreme earth analytics. In: 22nd International Conference on Extending Database Technology, EDBT 2019. Open Proceedings 690–693, URL http://nora.nerc.ac.uk/id/eprint/523287/.

Kussul, N., Lavreniuk, M., Skakun, S., Shelestov, A., 2017. Deep learning classification of land cover and crop types using remote sensing data. IEEE Geosci. Remote Sens. Lett. 14 (5), 778–782. http://dx.doi.org/10.1109/LGRS.2017.2681128.

Lee, J., Han, D., Shin, M., Im, J., Lee, J., Quackenbush, L.J., 2020. Different spectral domain transformation for land cover classification using convolutional neural networks with multi-temporal satellite imagery. Remote Sens. 12 (7), 1097. http://dx.doi.org/10.3390/rs12071097.

Liang, L., Wang, G., 2021. Efficient recurrent attention network for remote sensing scene classification. IET Image Process. 15 (8), 1712–1721. http://dx.doi.org/10.1049/ipr2.12139, arXiv:https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/ipr2.12139.

Liu, S., He, C., Bai, H., Zhang, Y., Cheng, J., 2020. Light-weight attention semantic segmentation network for high-resolution remote sensing images. In: IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium. pp. 2595–2598. http://dx.doi.org/10.1109/IGARSS39084.2020.9324723.

Lu, X., Sun, H., Zheng, X., 2019. A feature aggregation convolutional neural network for remote sensing scene classification. IEEE Trans. Geosci. Remote Sens. 57 (10), 7894–7906. http://dx.doi.org/10.1109/TGRS.2019.2917161.

Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P., 2016. Convolutional neural networks for large-scale remote-sensing image classification. IEEE Trans. Geosci. Remote Sens. 55 (2), 645–657. http://dx.doi.org/10.1109/TGRS.2016.2612821.

Mañas, O., Lacoste, A., Giro-i Nieto, X., Vazquez, D., Rodriguez, P., 2021. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9414–9423. http://dx.doi.org/10.1109/ICCV48922.2021.00928.

Maqueda, A.I., Loquercio, A., Gallego, G., García, N., Scaramuzza, D., 2018. Event-based vision meets deep learning on steering prediction for self-driving cars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, http://dx.doi.org/10.1109/CVPR.2018.00568.

Martini, M., Mazzia, V., Khaliq, A., Chiaberge, M., 2021. Domain-adversarial training of self-attention-based networks for land cover classification using multi-temporal sentinel-2 satellite imagery. Remote Sens. 13 (13), 2564. http://dx.doi.org/10.3390/rs13132564.

Md. Rafi, R.H., Tang, B., Du, Q., Younan, N.H., 2019. Attention-based domain adaptation for hyperspectral image classification. In: IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium. pp. 67–70. http://dx.doi.org/10.1109/IGARSS.2019.8898850.

Naushad, R., Kaur, T., Ghaderpour, E., 2021. Deep transfer learning for land use and land cover classification: A comparative study. Sensors 21 (23), http://dx.doi.org/10.3390/s21238083, URL https://www.mdpi.com/1424-8220/21/23/8083.

Perez, L., Wang, J., 2017. The effectiveness of data augmentation in image classification using deep learning. CoRR abs/1712.04621 arXiv:1712.04621.

Qian, Y., Zhou, W., Yan, J., Li, W., Han, L., 2015. Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. Remote Sens. 7 (1), 153–168. http://dx.doi.org/10.3390/rs70100153, URL https://www.mdpi.com/2072-4292/7/1/153.

Rahhal, M.M.A., Bazi, Y., Al-Hwiti, H., Alhichri, H., Alajlan, N., 2020. Adversarial learning for knowledge adaptation from multiple remote sensing sources. IEEE Geosci. Remote Sens. Lett. 1–5. http://dx.doi.org/10.1109/LGRS.2020.3003566.

Read, J., Pfahringer, B., Holmes, G., Frank, E., 2011. Classifier chains for multi-label classification, 85 (3). pp. 333–359. http://dx.doi.org/10.1007/s10994-011-5256-5.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham, pp. 234–241.

Schmitt, M., Hughes, L.H., Qiu, C., Zhu, X.X., 2019. SEN12MS – a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. IV-2/W7. pp. 153–160. http://dx.doi.org/10.5194/isprs-annals-IV-2-W7-153-2019.

Schmitt, M., Wu, Y.-L., 2021. Remote sensing image classification with the SEN12ms dataset. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. V-2-2021. pp. 101–106. http://dx.doi.org/10.5194/isprs-annals-V-2-2021-101-2021.

Scott, G.J., England, M.R., Starms, W.A., Marcum, R.A., Davis, C.H., 2017. Training deep convolutional neural networks for land–cover classification of high-resolution imagery. IEEE Geosci. Remote Sens. Lett. 14 (4), 549–553. http://dx.doi.org/10.1109/LGRS.2017.2657778.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision. ICCV, http://dx.doi.org/10.1109/ICCV.2017.74.

Sergeev, A., Balso, M.D., 2018. Horovod: fast and easy distributed deep learning in TensorFlow. arXiv:1802.05799.

Shao, J., Tang, L., Liu, M., Shao, G., Sun, L., Qiu, Q., 2020. BDD-Net: A general protocol for mapping buildings damaged by a wide range of disasters based on satellite imagery. Remote Sens. 12 (10), http://dx.doi.org/10.3390/rs12101670, URL https://www.mdpi.com/2072-4292/12/10/1670.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Srivastava, R.K., Greff, K., Schmidhuber, J., 2015. Highway networks. CoRR abs/1505.00387 arXiv:1505.00387.

Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., Beyer, L., 2021. How to train your ViT? Data, augmentation, and regularization in vision transformers. arXiv preprint arXiv:2106.10270.

Stivaktakis, R., Tsagkatakis, G., Tsakalides, P., 2019a. Deep learning for multilabel land cover scene categorization using data augmentation. IEEE Geosci. Remote Sens. Lett. 16 (7), 1031–1035. http://dx.doi.org/10.1109/LGRS.2019.2893306.

Stojnic, V., Risojevic, V., 2021. Self-supervised learning of remote sensing scene representations using contrastive multiview coding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1182–1191. http://dx.doi.org/10.1109/CVPRW53098.2021.00129.

Sumbul, G., Charfuelan, M., Demir, B., Markl, V., 2019. Bigearthnet: A large-scale benchmark archive for remote sensing image understanding. In: IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium. pp. 5901–5904. http://dx.doi.org/10.1109/IGARSS.2019.8900532.

Sumbul, G., Demir, B., 2019. A novel multi-attention driven system for multi-label remote sensing image classification. In: IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 5726–5729.

Sumbul, G., Demİr, B., 2020. A deep multi-attention driven approach for multi-label remote sensing image classification. IEEE Access 8, 95934–95946. http://dx.doi.org/10.1109/ACCESS.2020.2995805.

Sumbul, G., Kang, J., Kreuziger, T., Marcelino, F., Costa, H., Benevides, P., Caetano, M., Demir, B., 2020. BigEarthNet dataset with a new class-nomenclature for remote sensing image understanding. arXiv:2001.06372.

Sumbul, G., Ravanbakhsh, M., Demir, B., 2022. Informative and representative triplet selection for multi-label remote sensing image retrieval. IEEE Trans. Geosci. Remote Sens. 60, 1–11. http://dx.doi.org/10.1109/TGRS.2021.3124326.

Sumbul, G., de Wall, A., Kreuziger, T., Marcelino, F., Costa, H., Benevides, P., Caetano, M., Demir, B., Markl, V., 2021b. BigEarthNet-MM: A large scale multi-modal multi-label benchmark archive for remote sensing image classification and retrieval. arXiv:2105.07921.

Sumbul, G., de Wall, A., Kreuziger, T., Marcelino, F., Costa, H., Benevides, P., Caetano, M., Demir, B., Markl, V., 2021c. BigEarthNet-MM: A large-scale, multimodal, multilabel benchmark archive for remote sensing image classification and retrieval [Software and Data Sets]. IEEE Geosci. Remote Sens. Mag. 9 (3), 174–180. http://dx.doi.org/10.1109/MGRS.2021.3089174.

Sykas, D., Sdraka, M., Zografakis, D., Papoutsis, I., 2022. A sentinel-2 multiyear, multicountry benchmark dataset for crop classification and segmentation with deep learning. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 15, 3323–3339. http://dx.doi.org/10.1109/JSTARS.2022.3164771.

Talukdar, S., Singha, P., Mahato, S., Shahfahad, Pal, S., Liou, Y.-A., Rahman, A., 2020. Land-use land-cover classification by machine learning classifiers for satellite observations—A review. Remote Sens. 12 (7), http://dx.doi.org/10.3390/rs12071135, URL https://www.mdpi.com/2072-4292/12/7/1135.

Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V., 2019. MnasNet: Platform-aware neural architecture search for mobile. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2815–2823. http://dx.doi.org/10.1109/CVPR.2019.00293.

Tan, M., Le, Q., 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In: Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 97, PMLR, pp. 6105–6114, URL https://proceedings.mlr.press/v97/tan19a.html.

Tan, M., Pang, R., Le, Q.V., 2020. Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10781–10790. http://dx.doi.org/10.1109/CVPR42600.2020.01079.

Tang, X., Ma, Q., Zhang, X., Liu, F., Ma, J., Jiao, L., 2021. Attention consistent network for remote sensing scene classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14, 2030–2045. http://dx.doi.org/10.1109/JSTARS.2021.3051569.

Tian, Z., Wang, W., Tian, B., Zhan, R., Zhang, J., 2020. Resolution-Aware Network With Attention Mechanisms For Remote Sensing Object Detection. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. V-2-2020. Copernicus GmbH, pp. 909–916. http://dx.doi.org/10.5194/isprs-annals-V-2-2020-909-2020, URL https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-2-2020/909/2020/, ISSN: 2194-9042.

Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al., 2021. Mlp-mixer: An all-mlp architecture for vision. Adv. Neural Inf. Process. Syst. 34.

Tong, W., Chen, W., Han, W., Li, X., Wang, L., 2020a. Channel-attention-based DenseNet network for remote sensing image scene classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 13, 4121–4132. http://dx.doi.org/10.1109/JSTARS.2020.3009352.

Tong, X.-Y., Xia, G.-S., Lu, Q., Shen, H., Li, S., You, S., Zhang, L., 2020b. Land-cover classification with high-resolution remote sensing images using transferable deep models. Remote Sens. Environ. 237, 111322. http://dx.doi.org/10.1016/j.rse.2019.111322, URL https://www.sciencedirect.com/science/article/pii/S0034425719303414.

Vaswani, A., Bengio, S., Brevdo, E., Chollet, F., Gomez, A.N., Gouws, S., Jones, L., Kaiser, L., Kalchbrenner, N., Parmar, N., Sepassi, R., Shazeer, N., Uszkoreit, J., 2018. Tensor2Tensor for neural machine translation. CoRR abs/1803.07416 arXiv:1803.07416.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need. CoRR abs/1706.03762 arXiv:1706.03762.

Vincenzi, S., Porrello, A., Buzzega, P., Cipriano, M., Fronte, P., Cuccu, R., Ippoliti, C., Conte, A., Calderara, S., 2021. The color out of space: learning self-supervised representations for earth observation imagery. In: 2020 25th International Conference on Pattern Recognition. ICPR, pp. 3034–3041. http://dx.doi.org/10.1109/ICPR48806.2021.9413112.

Wang, C., Bai, X., Wang, S., Zhou, J., Ren, P., 2019a. Multiscale visual attention networks for object detection in VHR remote sensing images. IEEE Geosci. Remote Sens. Lett. 16 (2), 310–314. http://dx.doi.org/10.1109/LGRS.2018.2872355.

Wang, S., Chen, W., Xie, S.M., Azzari, G., Lobell, D.B., 2020a. Weakly supervised deep learning for segmentation of remote sensing imagery. Remote Sens. 12 (2), http://dx.doi.org/10.3390/rs12020207, URL https://www.mdpi.com/2072-4292/12/2/207.

Wang, Q., Liu, S., Chanussot, J., Li, X., 2019b. Scene classification with recurrent attention of VHR remote sensing images. IEEE Trans. Geosci. Remote Sens. 57 (2), 1155–1167. http://dx.doi.org/10.1109/TGRS.2018.2864987.

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q., 2020b. ECA-net: Efficient channel attention for deep convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 11531–11539. http://dx.doi.org/10.1109/CVPR42600.2020.01155.

Woo, S., Park, J., Lee, J.-Y., Kweon, I.S., 2018. CBAM: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision. ECCV.

Wu, Z.-Z., Wan, S.-H., Wang, X.-F., Tan, M., Zou, L., Li, X.-L., Chen, Y., 2020a. A benchmark data set for aircraft type recognition from remote sensing images. Appl. Soft Comput. 89, 106132. http://dx.doi.org/10.1016/j.asoc.2020.106132, URL https://www.sciencedirect.com/science/article/pii/S1568494620300727.

Wu, H., Zhao, S., Li, L., Lu, C., Chen, W., 2020b. Self-attention network with joint loss for remote sensing image scene classification. IEEE Access 8, 210347–210359. http://dx.doi.org/10.1109/ACCESS.2020.3038989.

Xia, G.-S., Hu, J., Hu, F., Shi, B., Bai, X., Zhong, Y., Zhang, L., Lu, X., 2017. AID: A benchmark data set for performance evaluation of aerial scene classification. IEEE Trans. Geosci. Remote Sens. 55 (7), 3965–3981. http://dx.doi.org/10.1109/TGRS.2017.2685945.

Yang, Y., Newsam, S., 2010. Bag-of-visual-words and spatial extensions for land-use classification. In: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. pp. 270–279. http://dx.doi.org/10.1145/1869790.1869829.

Ye, Y., Ren, X., Zhu, B., Tang, T., Tan, X., Gui, Y., Yao, Q., 2022. An adaptive attention fusion mechanism convolutional network for object detection in remote sensing images. Remote Sens. 14 (3), http://dx.doi.org/10.3390/rs14030516, URL https://www.mdpi.com/2072-4292/14/3/516.

Zagoruyko, S., Komodakis, N., 2016. Wide residual networks. CoRR abs/1605.07146 arXiv:1605.07146.

Zhang, C., Harrison, P.A., Pan, X., Li, H., Sargent, I., Atkinson, P.M., 2020. Scale sequence joint deep learning (SS-JDL) for land use and land cover classification. Remote Sens. Environ. 237, 111593. http://dx.doi.org/10.1016/j.rse.2019.111593.

Zhang, C., Pan, X., Li, H., Gardiner, A., Sargent, I., Hare, J., Atkinson, P.M., 2018. A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. ISPRS J. Photogramm. Remote Sens. 140, 133–144. http://dx.doi.org/10.1016/j.isprsjprs.2017.07.014.

Zhang, M.-L., Zhou, Z.-H., 2014. A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. 26 (8), 1819–1837. http://dx.doi.org/10.1109/TKDE.2013.39.

Zhao, W., Du, S., 2016. Learning multiscale and deep representations for classifying remotely sensed imagery. ISPRS J. Photogramm. Remote Sens. 113, 155–165. http://dx.doi.org/10.1016/j.isprsjprs.2016.01.004.

Zhao, W., Ivanov, I., Persello, C., Stein, A., 2020a. Building outline delineation: from very high resolution remote sensing imagery toO polygons with an improved end-to-end learning framework. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLIII-B2-2020, 731–735. http://dx.doi.org/10.5194/isprs-archives-XLIII-B2-2020-731-2020, URL https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLIII-B2-2020/731/2020/.

Zhao, Z., Li, J., Luo, Z., Li, J., Chen, C., 2020b. Remote sensing image scene classification based on an enhanced attention module. IEEE Geosci. Remote Sens. Lett. 18 (11), 1926–1930. http://dx.doi.org/10.1109/LGRS.2020.3011405.

Zhong, Z., Li, Y., Ma, L., Li, J., Zheng, W.-S., 2021. Spectral-spatial transformer network for hyperspectral image classification: A factorized architecture search framework. IEEE Trans. Geosci. Remote Sens. 1–15. http://dx.doi.org/10.1109/TGRS.2021.3115699.

Zhu, X.X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., Fraundorfer, F., 2017. Deep learning in remote sensing: A comprehensive review and list of resources. IEEE Geosci. Remote Sens. Mag. 5 (4), 8–36. http://dx.doi.org/10.1109/MGRS.2017.2762307.