applied sciences

MDPI

Article

Cross-Domain Transfer Learning for Natural Scene Classification of Remote-Sensing Imagery

Muhammad Akhtar [1], Iqbal Murtza [1], Muhammad Adnan [2,*] and Ayesha Saadia [1]

[1] Faculty of Computing & AI, Air University, Islamabad 44000, Pakistan
[2] Department of Technology and Safety, UiT The Arctic University of Norway, 9019 Tromsø, Norway
* Correspondence: muhammad.adnan@uit.no

Abstract: Natural scene classification, which has potential applications in precision agriculture, environmental monitoring, and disaster management, poses significant challenges due to variations in the spatial resolution, spectral resolution, texture, and size of remotely sensed images of natural scenes on Earth. For such challenging problems, deep-learning-based algorithms have demonstrated amazing performances in recent years. Among these methodologies, transfer learning is a useful technique which employs the learned features already extracted from the pre-trained models from large-scale datasets for the problem at hand, resulting in quicker and more accurate models. In this study, we deployed cross-domain transfer learning for the land-cover classification of remotely sensed images of natural scenes. We conducted extensive experiments to measure the performance of the proposed method and explored the factors that affect the performance of the models. Our findings suggest that fine-tuning the ResNet-50 model outperforms various other models in terms of the classification accuracy. The experimental results showed that the deployed cross-domain transfer-learning system achieved outstanding (99.5% and 99.1%) accurate performances compared to previous benchmarks on the NaSC-TG2 dataset with the final tuning of the whole structure and only the last three layers, respectively.

Keywords: natural scene classification; land cover; deep learning; remote sensing; convolutional neural networks; transfer learning

Citation: Akhtar, M.; Murtza, I.; Adnan, M.; Saadia, A. Cross-Domain Transfer Learning for Natural Scene Classification of Remote-Sensing Imagery. *Appl. Sci.* **2023**, *13*, 7882. https://doi.org/10.3390/app13137882

Academic Editor: Andrea Prati

Received: 26 May 2023
Revised: 29 June 2023
Accepted: 3 July 2023
Published: 5 July 2023

1. Introduction

Remote-sensing technology has made it possible to gather and analyze data from the Earth's surface in an effective manner. Because of the rising demand for accurate image classification in many areas, the natural scene classification of remotely sensed images has become a prominent study area. In this domain, remote-sensing image classification [1] is an important problem in detecting and mapping various forms of land cover on the Earth's surface and its manual labeling is a tedious task. This is the reason why the automated identification of natural landscapes in remote-sensing images has gained substantial attention from the research community [1–5].

Recently, deep learning has emerged as a powerful tool for the automated extraction of relevant features and their exploitation. Deep-learning-based algorithms have outperformed traditional approaches in various computer vision applications, including natural scene classification [2]. Transfer learning is a prominent deep-learning method in which pre-trained models on big datasets are reused for new tasks with smaller datasets. Transfer learning may take pre-trained model information and apply it to new tasks, resulting in quicker and more accurate models [6].

Remote-sensing datasets vary in spatial resolution, spectral resolution, number of visual classes, number of images per class, and total number of images in a dataset [3]. Despite a substantial amount of study regarding this subject, suggested models on one dataset may not perform well when evaluated on other datasets because of a number of

reasons, for example, the base dataset size, variation, structure of deep learning, etc. This is why the base (pre-trained) model is of critical importance in transfer learning.

In this paper, we applied transfer learning on a NaSC-TG2 dataset [7] using pre-trained deep-learning models. Previously, a VGG-16 model showed outstanding performance (overall accuracy: 89.59%) on a NaSC-TG2, setting a benchmark for further studies. The architecture of a VGG-16 model [8] is shown in Figure 1.

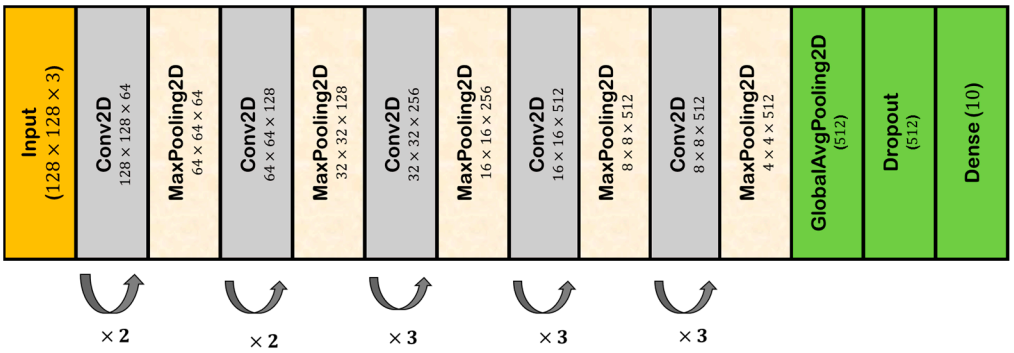

Figure 1. VGG-16 Architecture.

2. Relevant Work

In this section, we review the relevant work on classifying remote-sensing images into natural scenes using traditional machine learning, deep learning, and convolutional neural networks. We present an overview of the literature on these topics and discuss how our work builds upon and extends previous research.

2.1. Remote-Sensing Image Scene Classification

Many academics have worked in recent years on using various strategies to increase the accuracy of natural scene classification. Traditional classification approaches, such as supervised and unsupervised classification, are frequently employed. To properly classify remote-sensing images, supervised classification techniques such as maximum likelihood classification and support vector machines (SVMs) require labeled training sets. Because it can handle big datasets and complicated feature spaces with high accuracy, SVM has been proved to be a useful approach for the natural scene categorization of remote-sensing images [9].

Natural scene classification of remotely sensed images has been widely studied in recent years to improve classification accuracy using traditional machine-learning techniques. Among these methods, deep-learning techniques, especially CNNs with transfer learning, have shown promising results for the natural scene classification of remote-sensing images. Attention mechanisms and feature-fusion methods have also shown significant improvements in classification performance.

2.2. Deep Learning

LeCun et al. [10] introduced the concept of deep learning and its applications in various domains, including computer vision. The authors discussed the fundamental principles of deep learning and its potential in solving complex problems by automatically learning hierarchical representations from data. Cheng et al. [4] conducted a comprehensive survey on remotely sensed image scene categorization using deep-learning algorithms. The authors reviewed the difficulties encountered in remote-sensing picture scene categorization and offered an outline of the various deep-learning approaches used in this sector. They also reviewed benchmark datasets used to evaluate the performance of classification systems and highlighted future research prospects in this field.

Cheng et al. [11] provided a benchmark and up-to-date review of remote-sensing picture categorization. They talked about several classification strategies, such as classic

ML and DL methods. The authors assessed the performance of different approaches on benchmark datasets and discussed their advantages and disadvantages. The AID dataset, developed by Xia et al. [12], serves as a baseline for testing the performance of aerial scene categorization systems. The authors thoroughly presented the dataset, including scene categories and picture attributes, and emphasized its significance in furthering research in aerial scene categorization.

Zhou Q. et al. [13] developed a flexible segmentation graph-based multi-dimensional contextual method for scene annotation. The authors proved the method's performance on multiple datasets, demonstrating its potential for properly categorizing scenes based on contextual information. Zhou B. et al. [14] presented a method for learning deep features that are discriminative for object localization in images. The authors described how their approach enables the network to focus on relevant object details, leading to improved object localization performance.

He et al. [15] presented deep residual networks and established the notion of residual learning. Their research proved that by exploiting residual connections, deeper networks may be trained more successfully, resulting in superior performance in image classification tasks. Bu et al. [16] suggested a method for scene parsing that makes use of inference-embedded deep networks. The authors described how their method incorporates inference steps within the network architecture, enabling efficient scene parsing. They evaluated the effectiveness of their approach on the Pascal VOC dataset, demonstrating accurate scene-parsing results.

Pohlen et al. [17] proposed using full-resolution residual networks to separate street scenes for semantic segmentation. Their strategy outperformed competitors on the Cityscapes dataset, demonstrating the utility of full-resolution processing in semantic segmentation tasks. Tombe et al. [18] introduced an adaptive deep co-occurrence feature-learning strategy for remote-sensing scene categorization based on classifier fusion. The scientists revealed how they used deep-learning features and co-occurrence matrices to capture both spectral and spatial information in remote-sensing pictures, resulting in enhanced classification accuracy.

Boualleg et al. [19] suggested a technique for remote-sensing scene classification combining CNN-based features and a deep forest classifier. The authors highlighted the advantages of their approach, including its ability to handle high-dimensional data and its efficiency in training and classification. Zhu et al. [20] investigated the application of generative adversarial networks (GANs) for visual scene categorization in remote sensing. The authors suggested a GAN-based framework that uses the adversarial training process to produce synthetic samples, which are then mixed with actual examples to improve the classification model's diversity and generalization capabilities.

Fang et al. [21] addressed the challenge of limited labeled data in remotely sensed image scene classification by proposing a semi-supervised learning approach. The authors developed a co-training algorithm that utilizes both labeled and unlabeled data to increase the classification accuracy of the model. Xu et al. [22] proposed an end-to-end ET-GSNet solution for remote-sensing image scene classification. It combines the strengths of Vision Transformer (ViT) and ResNet18 through knowledge distillation. The proposed method outperforms state-of-the-art algorithms in classification performance on remote-sensing databases. In many ways, it also demonstrates excellent generality for a wide range of occupations.

2.3. Convolutional Neural Networks

Assigning a semantic label to an image based on its content is known as natural scene classification, and it is a challenging task because of the variety and complexity of the remotely sensed images. According to Kaul et al. [5], deep-learning approaches, notably CNNs, have shown amazing performance in the natural scene classification of remote-sensing data. Several CNN architectures and approaches, including AlexNet, VGG-16, ResNet, transfer learning, and attention processes, have been suggested and

used for this problem. These strategies have significantly increased classification accuracy while decreasing the quantity of training data necessary for natural scene categorization in remote-sensing pictures.

AlexNet, introduced by Kryszewsky et al. [6], is one of the oldest and most prominent CNN architectures for image classification. AlexNet won the ImageNet challenge and outperformed the competition on a variety of image classification benchmarks. Since then, AlexNet has been widely employed in a variety of applications, including natural scene categorization in remote-sensing data. Zhao G et al. [23], for example, provided a multi-sensor data-fusion framework for natural scene categorization in which AlexNet was utilized as the classification model to combine information derived from various sensors.

Several deeper and more complicated CNN architectures have been suggested and utilized for natural scene categorization in remote-sensing images since AlexNet. For example, VGG-16 and ResNet have been employed to increase the classification accuracy on various datasets. VGG-16, which features 16 convolutional layers and a considerably deeper network architecture than AlexNet, has demonstrated an enhanced performance on natural scene categorization tests. To solve the issue of disappearing gradients in deep networks, ResNet incorporated residual connections between layers. This method allows the network to be trained in greater depth without losing performance. Simonyan et al. [8] presented VGG-Net, a very deep convolutional neural network architecture. The authors described the network's architecture, which consists of multiple convolutional layers, achieving excellent performance on the ImageNet dataset and establishing a new benchmark in image classification [8,15].

Li et al. [24] examined deep-learning strategies for remotely sensed image scene categorization, with a special focus on feature extraction and classification algorithms. To capture both spatial and temporal information in remote-sensing data, the authors devised a hybrid deep-learning architecture that integrates CNNs and RNNs. Dai et al. [25] built on their earlier work by developing a unique attention-based deep-learning model for remote-sensing picture scene categorization. The scientists added an attention mechanism to the CNN design to selectively focus on relevant portions of the input picture, increasing the model's discriminative strength. Ghadi et al. [26] suggested a feature-fusion method for classifying remotely sensed images that incorporated spectral and textural data using a CNN. The suggested technique outperformed standard feature extraction methods in terms of performance.

Unfortunately, in these available techniques, transfer learning is not exploited well because of the scarcity of pre-trained models. If used rarely, it results in constrained performance in-domain transfer learning. Unfortunately, the employment of cross-domain transfer learning is also rare because of the risk of low performance in nature scene classification. Here, we go for it.

3. Materials and Methods

3.1. NaSC-TG2 Dataset

Natural Scene Classification with Tiangong-2 (NaSC-TG2) dataset [7] was used to fine-tune a pre-trained ResNet-50 model. The NaSC-TG2 dataset contains a total of 20,000 images of 128 × 128 size with 10 classes for remotely sensed image scenes. Sample images for different classes are displayed in Figure 2. The NaSC-TG2 dataset addresses the limitations of existing remote-sensing image datasets by offering several distinct properties. Firstly, it provides a large-scale dataset that overcomes the shortage of labeled scene images in remote sensing. This allows for more effective training of complex deep-learning networks, making it a valuable resource for the remote-sensing community. The dataset also ensures a balanced distribution of scenes, contributing to improved network training and evaluation. Secondly, NaSC-TG2 exhibits large intra-class differences and high inter-class similarity, mimicking the complex and variable conditions found on the Earth's surface. This challenges classification methods to be more robust and generalize well to accurately classify scene images. Thirdly, the dataset includes natural scenes with novel spatial scales

and imaging performance, setting it apart from existing datasets that primarily focus on artificial landscapes. This diversity of natural scenes enables comprehensive algorithm verification and analysis, particularly in the field of natural scene classification. Additionally, NaSC-TG2 includes 14-band multi-spectral scene images alongside true-color RGB images, providing valuable data for high-dimensional scene image classification research. Overall, the NaSC-TG2 dataset offers a large-scale, diverse, and comprehensive benchmark for remote-sensing scene classification methods, addressing the limitations of previous datasets and promoting advancements in the field.

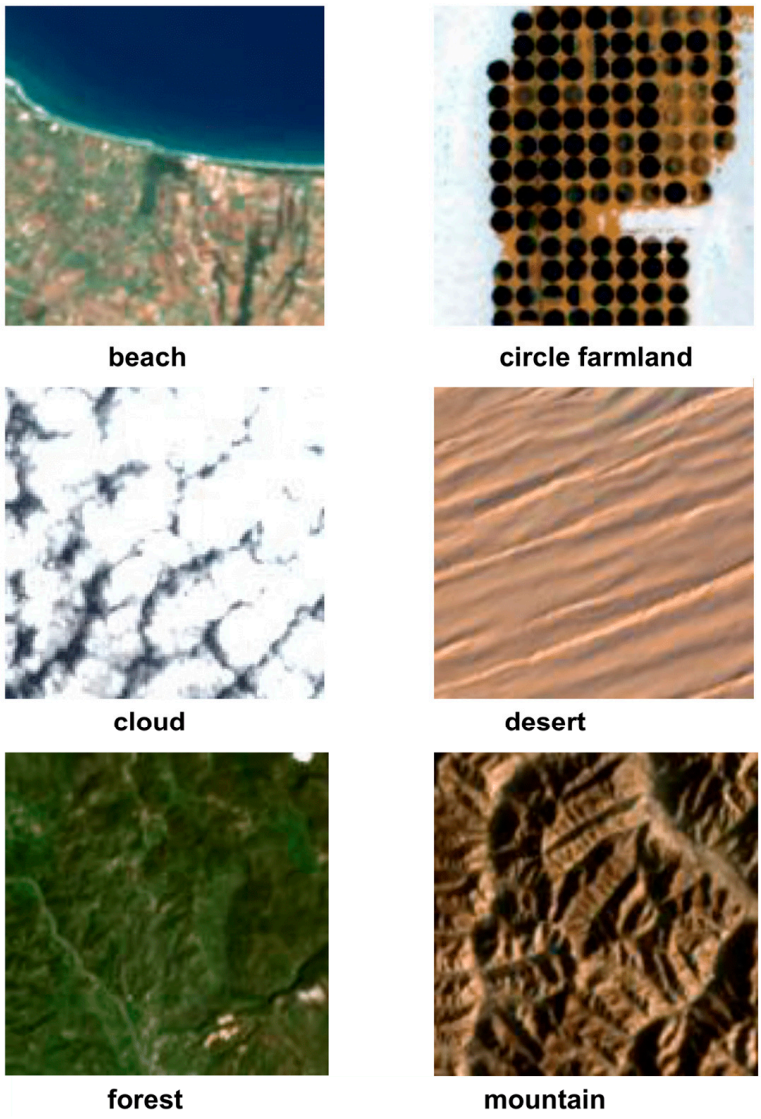

Figure 2. *Cont.*

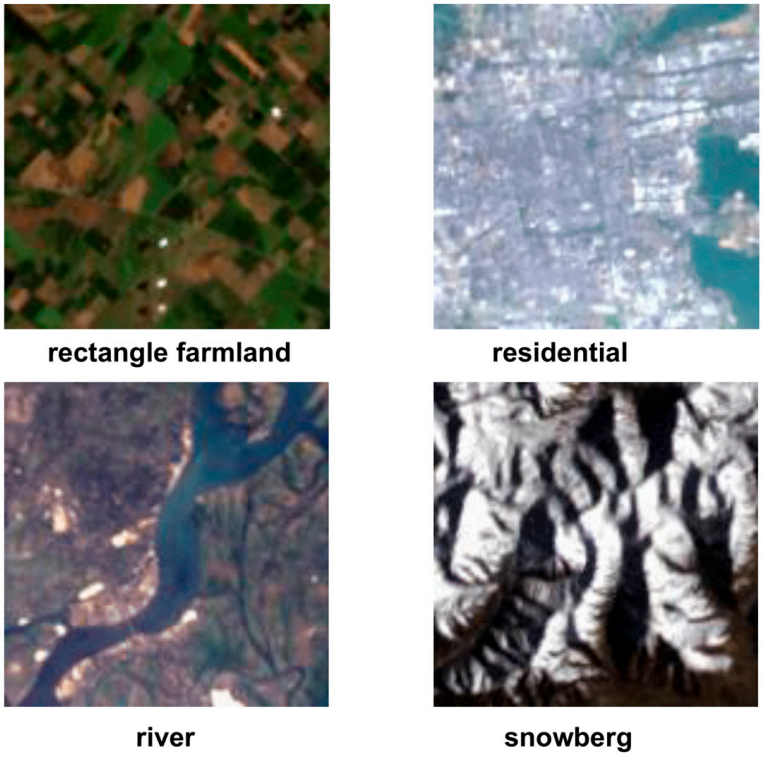

Figure 2. NaSC-TG2 samples.

The number of images for each class are equally distributed, as shown in Figure 3. This dataset is challenging because of its high intra-class variations (see Figure 4) and inter-class similarities (see Figure 5). All images are obtained by wideband imaging spectrometer from the Tiangong-2 satellite. This contains both RGB and multi-spectral images but for this study, only RGB images were used. The overall benchmark accuracy achieved on this dataset was 89.59% using the VGG-16 model in an 80:20 train–test split.

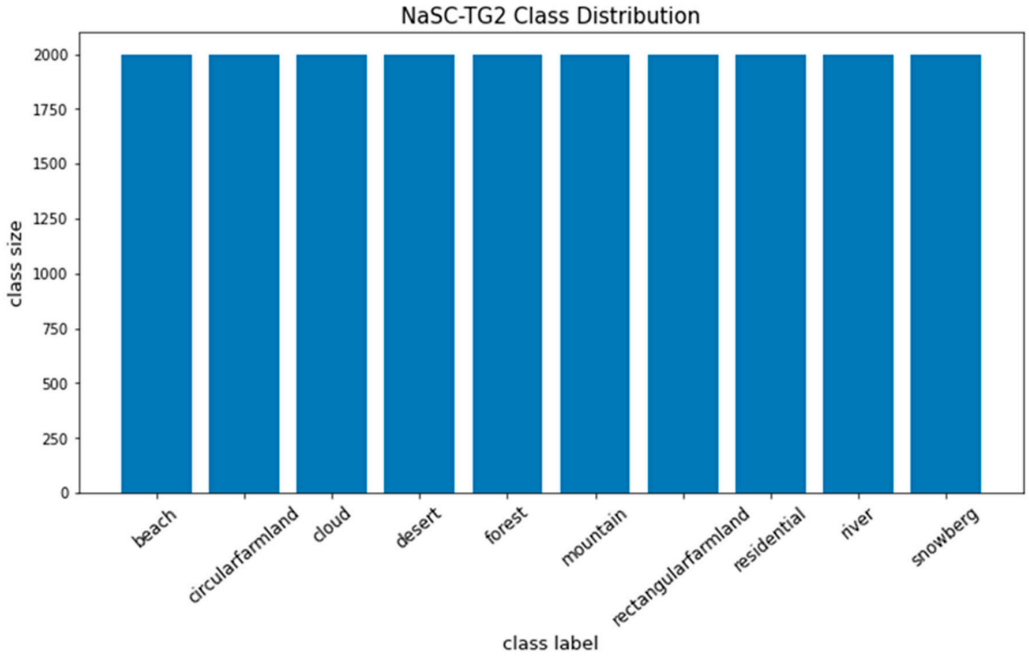

Figure 3. NaSC-TG2 Class Distribution.

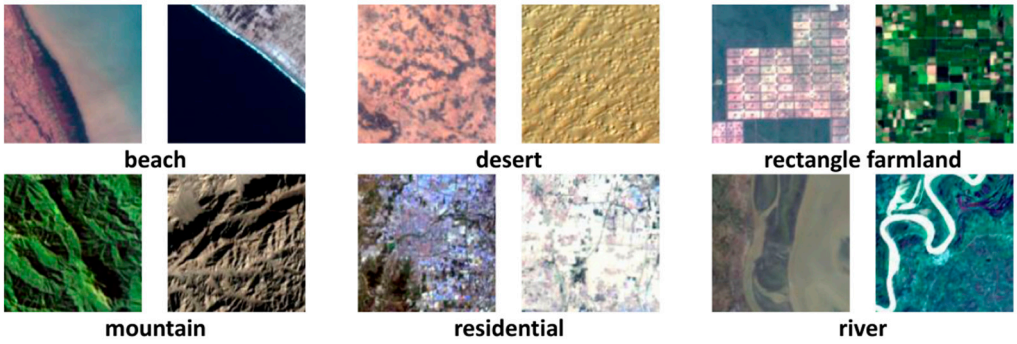

Figure 4. Large Intra-Class Diversity [7].

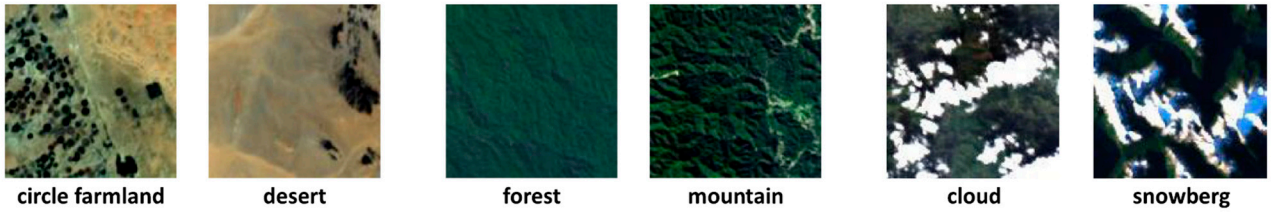

Figure 5. Small Inter-Class Similarity (**a**) Similar structural distributions between different classes. (**b**) Similar colors between different classes. (**c**) Similar objects between different classes [7].

3.2. Proposed Framework for Natural Scene Classification

Transfer learning with pre-trained "deep-learning" architectures may greatly increase natural scene categorization job performance by utilizing information from pre-trained models and fine-tuning them with fresh data, especially when the dataset is restricted.

Our proposed framework is shown in Figure 6.

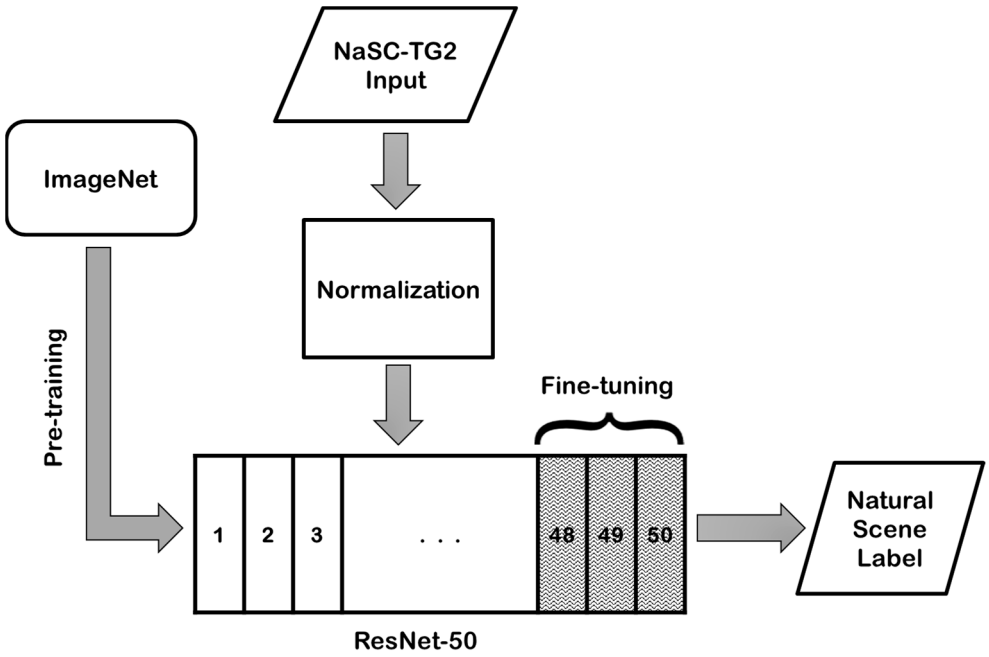

Figure 6. Structure of cross-domain ResNet-50 for natural scene classification wherein, model pre-trained on ImageNet is fine-tuned on NaSC-TG2 dataset.

The following methods are included in the proposed technique for natural scene classification using pre-trained deep-learning architectures:

3.2.1. Data Pre-Processing

This process included tasks such as data augmentation, normalization, and train–test splitting. Before training the deep-learning model, we performed the pre-processing.

Data Augmentation: We used data augmentation to expand the size of the dataset while reducing overfitting. We randomly applied transformations such as rotation, scaling, and flipping to the images.

Normalization: We normalized the image pixel values to have a zero mean and unit variance. This contributed to faster convergence during training.

Splitting the Dataset: The dataset was separated into training and testing sets with an 80:20 train–test split. Training data were further divided into training and validation subsets. While training the model, the validation set was used for hyper-parameter tuning and to avoid overfitting, and the testing set was used to calculate the model's overall performance.

3.2.2. Pre-Trained Model Selection

We deployed ResNet-50 and VGG-19 model architectures for cross-domain transfer-learning experiments. ResNet-50, as its name suggests, comprises 50 layers and it contains residual blocks which add input to the output of the block with the help of skip connections. It was introduced by Microsoft Research [15] in 2015 and has since become one of the most widely used and influential deep-learning models. The following are the key features of the ResNet-50 model.

Residual Learning: ResNet-50 came up with the idea of residual learning [15,27], which solved the vanishing gradient problem in deep neural networks. The vanishing gradient problem refers to the observation that as deep networks are trained, their performance starts saturating and then degrades rapidly. Residual learning tackles this issue by introducing skip connections (shortcut connections) which enable the network to learn residual representations rather than learning underlying representations directly. These skip connections (see Figure 7) enable the gradients to pass through the network directly, addressing the vanishing gradient problem and facilitating the training of extremely deep networks.

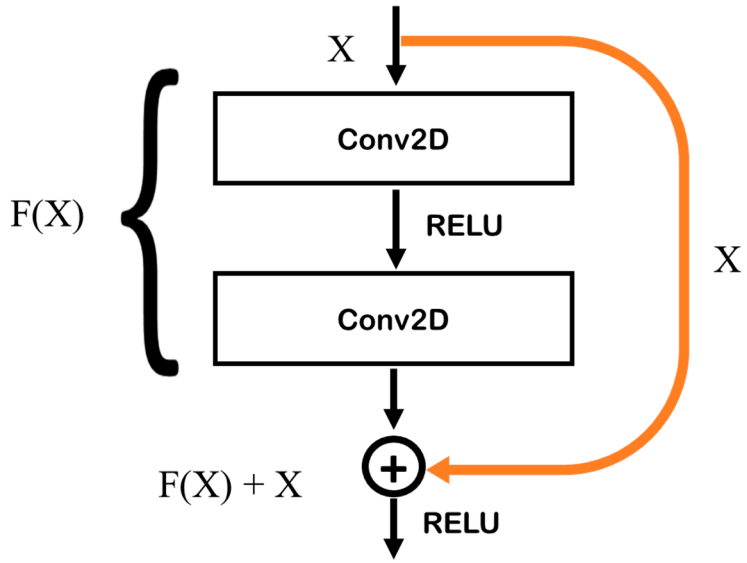

Figure 7. Residual Learning Building Block [15].

Architecture: ResNet-50 is comprised of different layers including convolutional layers, batch normalization layers, pooling layers, and finally, dropout and fully connected layers. The architecture is characterized by residual blocks, which are comprised of multiple convolutional layers with skip connections. Each residual block contains a shortcut

connection that skips one or more layers and merges the input directly with the output of the block, allowing the network to learn residual representations. The architecture gradually reduces spatial dimensions while increasing the number of channels, enabling the network to capture hierarchical features at different scales. We used the following model architecture (see Figure 8) for training on the NaSC-TG2 dataset.

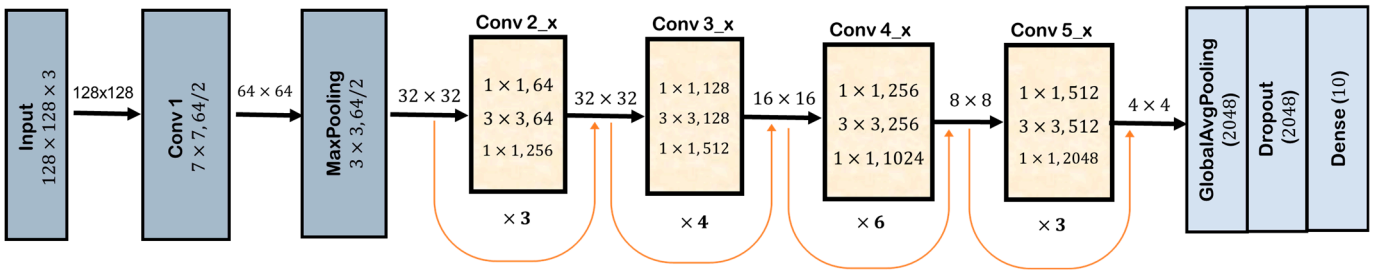

Figure 8. ResNet-50 Architecture.

Pre-training on ImageNet: ResNet-50, like other deep-learning models, benefits from pre-training on large-scale datasets. ResNet-50 is initialized with weights pre-trained on the ImageNet dataset. ImageNet is a large-scale dataset which contains 1.4 million labeled images belonging to 1000 classes. Pre-training on ImageNet allows the model to learn rich and generalizable features which are transferable to various computer vision tasks with limited labeled data.

3.2.3. Fine-Tuning Pre-Trained Model

The suggested technique for the natural scene classification of remote-sensing images involves using the pre-trained ResNet-50 network for feature extraction, followed by fine-tuning a new fully connected layer for the classification purpose. ResNet-50 is a widely used deep-learning architecture that has demonstrated cutting-edge performance on a variety of image classification datasets including ImageNet.

To put the suggested strategy into action, we first obtained the NaSC-TG2 dataset, which covers ten different types of natural landscapes, such as mountain, forest, beach, circular farmland, and residential, etc. We normalized the pixel values of the images to obtain zero mean and unit variance. The pre-trained ResNet-50 model was then loaded and its top layer was removed. We added new global average pooling, dropout, and a fully connected layer with 10 nodes for the classification of images from the NaSC-TG2 dataset. For the target task, we froze all the convolutional layers in ResNet-50 and only trained the fully connected layers added after the convolutional layers structure. The revised ResNet-50 model was then fine-tuned on the NaSC-TG2 dataset with the Adam optimizer at a 0.0001 learning rate and 32 batch size. To compute the difference between the anticipated and actual class labels, we employed a "categorical cross-entropy" loss function.

We conducted a number of experiments to evaluate the performance of the deployed models. Finally, we compared the proposed method's performance to that of existing cutting-edge techniques for natural scene classification, such as traditional machine-learning and deep-learning techniques for remote-sensing image classification. Overall, the proposed method combined the advantages of transfer learning and residual learning to achieve a high accuracy on the natural scene classification task, which has significant uses in applications such as environment monitoring, disaster management, urban planning, and protection of natural habitats. Finally, the performance of the fine-tuned model was evaluated using a set of testing images. This is typically performed by comparing the classified image to a reference image that has been collected from the same area.

3.3. Evaluation Metrics

The following metrics were used for evaluating our transfer-learning-based framework for natural scene classification.

3.3.1. Confusion Matrix

A confusion matrix is a tabular representation of actual values against predicted values for each class in the classification problem. It shows how many true and false predictions a model has produced for each class.

The confusion matrix's four types of entries are the following:

- True Positives (TP)
- False Positives (FP)
- True Negatives (TN)
- False Negatives (FN)

The number of true positives (TP) reflects the number of instances of a class that were correctly predicted, whereas the number of false positives (FP) represents the number of instances that were incorrectly predicted as belonging to that class. Similarly, true negatives (TN) are examples that were properly projected to be outside of that class, whereas false negatives (FN) are instances that were wrongly predicted to be outside of that class. We can compute other evaluation metrics by using entries from the confusion matrix.

3.3.2. Accuracy

The accuracy of a model is measured by the fraction of all true predictions by all predictions of the model. The formula to compute the accuracy is represented below in Equation (1).

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{TN} + \text{FN}) \quad (1)$$

It is a standard measure for evaluating classification tasks. However, accuracy is not always a reliable metric, especially in scenarios where the class distribution is imbalanced. For example, in a cancer diagnosis dataset where only 1% of the instances are cancerous, a model that predicts all instances as non-cancerous would still have a high accuracy rate.

3.3.3. Precision

Precision is an evaluation metric which calculates the proportion of true positive predictions made by a model out of all positive predictions. The formula to compute the precision of a classifier is given below in Equation (2).

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (2)$$

Precision is the number of instances of a class that were correctly predicted as a member of that class divided by the total number of instances predicted to belong to that class. Precision is a very critical measure especially in a situation where false positives are desired to be minimized. For example, in medical diagnosis, it is better to have low false positive rates to avoid unnecessary treatments. On the other hand, a high false negative rate is more acceptable in this case because it may result in missed diagnoses that can be corrected later.

3.3.4. Recall

The fraction of accurate positive predictions out of all real positive cases is referred to as recall. It is also called sensitivity or the true positive rate (TPR) of a model. The formula to compute recall is given below in Equation (3).

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (3)$$

It assesses how effectively a model can accurately detect all positive cases. Recall is a critical measure in a case where false negatives are more costly than false positives. For

example, in spam email classification, it is important to correctly identify all spam emails, even if some legitimate emails are classified as spam.

3.3.5. F1 Score

The F1 Score is an evaluation metric that takes the harmonic mean of precision and recall. The formula to compute the F1 score is given below in Equation (4):

$$\text{F1 Score} = (2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}) \quad (4)$$

The F1 Score is a trade-off between precision and recall, and it is more useful when the two measures diverge. For instance, if precision is high but recall is low, the F1 Score will be lower than the precision number, indicating that recall may be improved. Conversely, if recall is high but precision is low, the F1 Score will be lower than the recall value, showing that precision can be improved by minimizing false positives.

4. Results and Discussion

This section presents findings based on the suggested methodology described in the previous section. The models were trained on all classes simultaneously contrary to the contemporary many one-versus-all classifications. Two deep-learning pre-trained models, ResNet50 and VGG19, were used for transfer learning in the land-cover classification of remotely sensed natural scenes. We trained both models with three different modes. In the first scenario, we trained the whole structure, initializing it with pre-trained weights. In the second scenario, we used these networks as feature extractors and only trained the fully connected layers for classification. We trained only the last three layers and the parameters for the rest of the network were frozen. Finally, we trained these models from scratch, initializing the models with random weights instead of using pre-trained weights. This helped us to judge the effectiveness of the transfer-learning technique for remote-sensing image scene classification. In this way, we performed these experiments for the comparison of different training modes. We performed all experiments with 100 epochs and a learning rate of 0.0001. The train–test split was 80:20 and the training set was further divided into training and validation subsets with the same ratio. This means that 12,800 images for training, 3200 images for validation, and 4000 images for testing were used in all experiments.

4.1. Fine-Tuning the Entire Pre-Trained Model

To investigate the effect of all the weights of the entire pre-trained models on ImageNet, we fine-tuned all the weights of the entire pre-trained models on the dataset NaSC-TG2. For this, we used ImageNet weights as the starting point. Both ResNet-50 and VGG-19 were trained for 100 epochs and the plotted graphs of the model accuracy for training vs. validation are reported in Figures 9 and 10. The graph in Figure 9 shows that ResNet-50 started after a few epochs. This indicates that the pre-trained model has extensively learned abstract and complex features during its pre-training. While in Figure 10, the graph shows that VGG-19 does not converge in the early epochs and many fluctuations were observed during its training.

After training and evaluation, ResNet-50 and VGG-19 achieved an impressive overall accuracy of 99.50% and 98.02%, respectively. This was a significant improvement over the performance benchmarks for this dataset. Detailed results for each class using the ResNet-50 model are shown in Table 1.

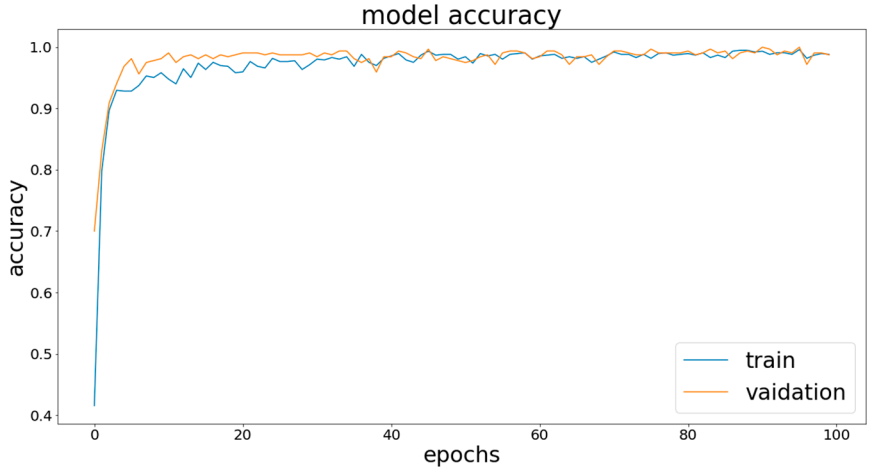

Figure 9. Training vs. validation accuracy curve during fine-tuning of entire ResNet-50 model.

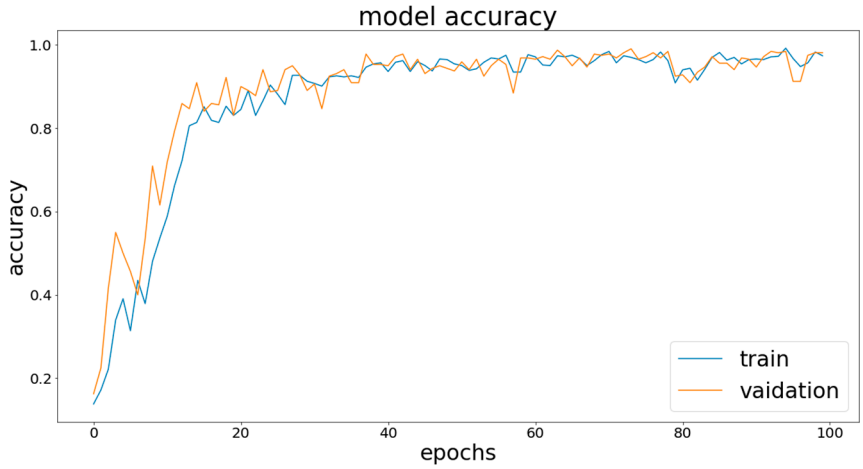

Figure 10. Training vs. validation accuracy curve during fine-tuning of entire VGG-19 model.

Table 1. Results for ResNet-50 when Entire Model is fine-tuned on NaSC-TG2.

Class	Recall	Precision	Accuracy	F1 Score
Beach	0.9950	0.9975	0.9992	0.9962
Circular farmland	1.0000	0.9975	0.9998	0.9988
Cloud	0.9975	0.9901	0.9988	0.9938
Desert	0.9900	1.0000	0.9990	0.9950
Forest	0.9975	0.9975	0.9995	0.9975
Mountain	0.9975	0.9876	0.9985	0.9925
Rectangular farmland	1.0000	1.0000	1.0000	1.0000
Residential	0.9975	0.9901	0.9988	0.9938
River	0.9900	0.9925	0.9982	0.9912
Snowberg	0.9850	0.9975	0.9982	0.9912
Overall	99.50%	99.50%	99.50%	99.50%

Looking at the confusion matrix in Figure 11, we can observe that the model performed admirably in all of the 10 classes, with a perfect performance (with no false negatives) in some, e.g., circular farmland and rectangular farmland and an almost perfect performance in others, e.g., beach, cloud, forest, mountain, and residential, etc., with only one or two

false negatives (FN). The snowberg class had six false negatives which was highest recorded FN score for any class. The highest false positives (FP) score was five, for the mountain class.

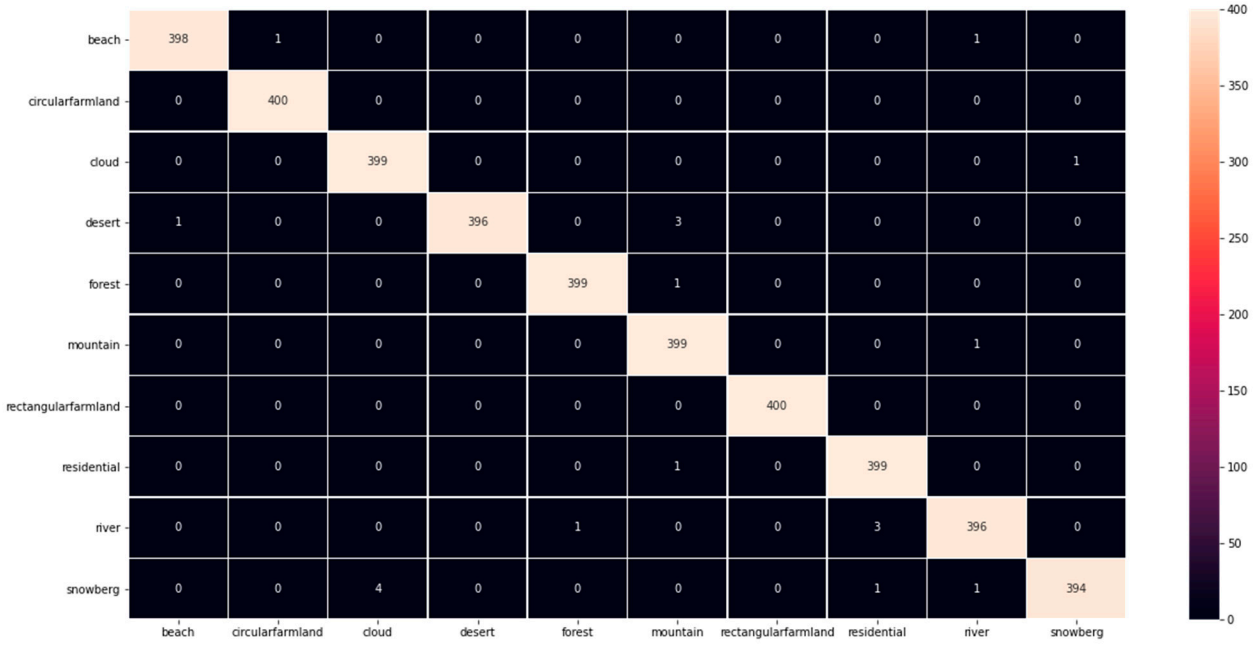

Figure 11. Confusion matrix for fine-tuning entire model of ResNet-50.

Overall, fine-tuning the entire model results in the highest accuracy as it optimizes the weights of the whole network according to the targeted domain. This is because we fine-tuned all the layers including the initial layers which capture low-level features and the final layers which capture high-level features. Since low-level features are similar for all types of images, only the final layers were adapted to the task of natural scene classification.

4.2. Fine-Tuning Only Last Three Layers of the Pre-Trained Model

In this case, we only trained the last three layers of the ResNet-50 model on the NaSC-TG2 dataset using ImageNet weights as the starting point. We froze all previous layers and those layers acted as the feature extractor. The graphs of the model accuracy have been plotted for training vs. validation during the training of the last three layers of both ResNet-50 and VGG-19. These graphs are shown in Figures 12 and 13. By analyzing these graphs, it is evident that the training behavior of these models was quite similar to the previous experiment where we fine-tuned the entire models. We observed that the training of the ResNet-50 model was saturated after a few epochs while the VGG-19 model was slowly saturated with a lot of fluctuations along the way.

After the training and evaluation, as shown in Figure 13, we achieved an overall accuracy of 99.10% and 97.28% for ResNet-50 and VGG-19, respectively. These results were slightly lower than in the previous case where the entire model was fine-tuned. However, these results were quite impressive because all the feature extraction parts of the network were frozen and only the classification part was fine-tuned on the NaSC-TG2 dataset. Detailed results for ResNet-50 indicating the evaluation metrics of the precision, recall, accuracy, and F1 score for each class in the dataset are shown in Table 2. We observed similar overall values for each evaluation measure because our dataset was evenly distributed among all the classes.

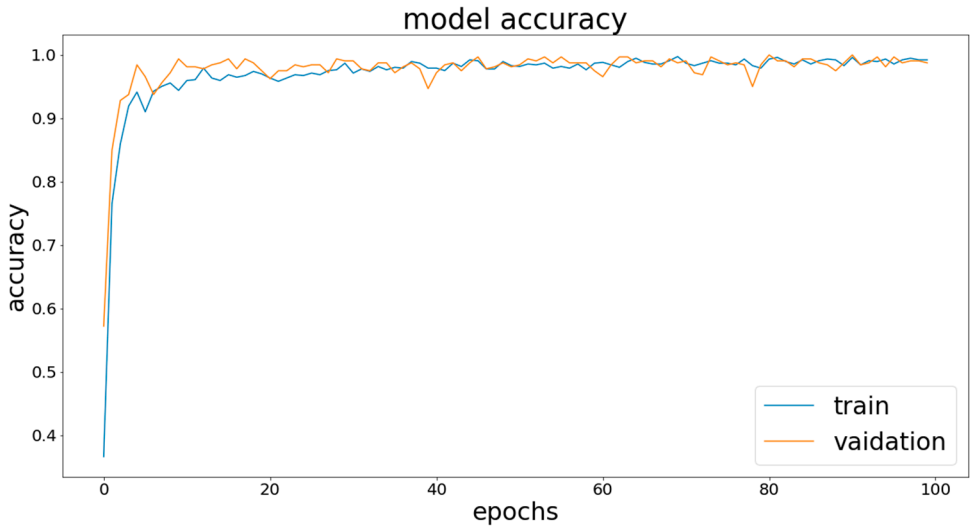

Figure 12. Training vs. validation accuracy curve during fine-tuning of last 3 layers of ResNet-50.

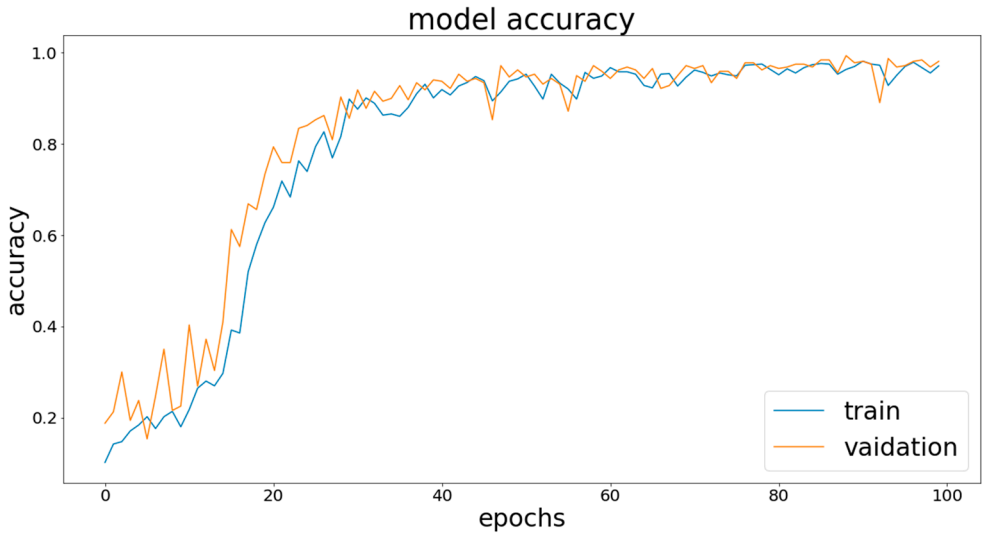

Figure 13. Training vs. validation accuracy curve during fine-tuning of last 3 layers of VGG-19.

Table 2. Last 3 layers of ResNet-50 Fine-tuned with NaSC-TG2.

Class	Recall	Precision	Accuracy	F1 Score
Beach	0.9925	1.0000	0.9992	0.9962
Circular farmland	0.9950	0.9925	0.9988	0.9938
Cloud	0.9975	0.9901	0.9988	0.9938
Desert	0.9950	0.9950	0.9990	0.9950
Forest	1.0000	1.0000	1.0000	1.0000
Mountain	0.9825	0.9949	0.9978	0.9887
Rectangular farmland	0.9800	0.9899	0.9970	0.9849
Residential	0.9950	0.9925	0.9988	0.9938
River	0.9800	0.9631	0.9942	0.9715
Snowberg	0.9925	0.9925	0.9985	0.9925
Overall	99.10%	99.11%	99.10%	99.10%

Figure 14 displays the confusion matrix for the fine-tuning of the last three layers of ResNet-50, and we can observe that the model performed admirably in all of the 10 classes, with no false negatives (FN) at all in the forest class, whereas the beach, circular farmland, cloud, and desert class have three false negatives. Rectangular farmland had eight false negatives which were misclassified as river class. River class had 15 false positives (FP) which was the highest FP for any class.

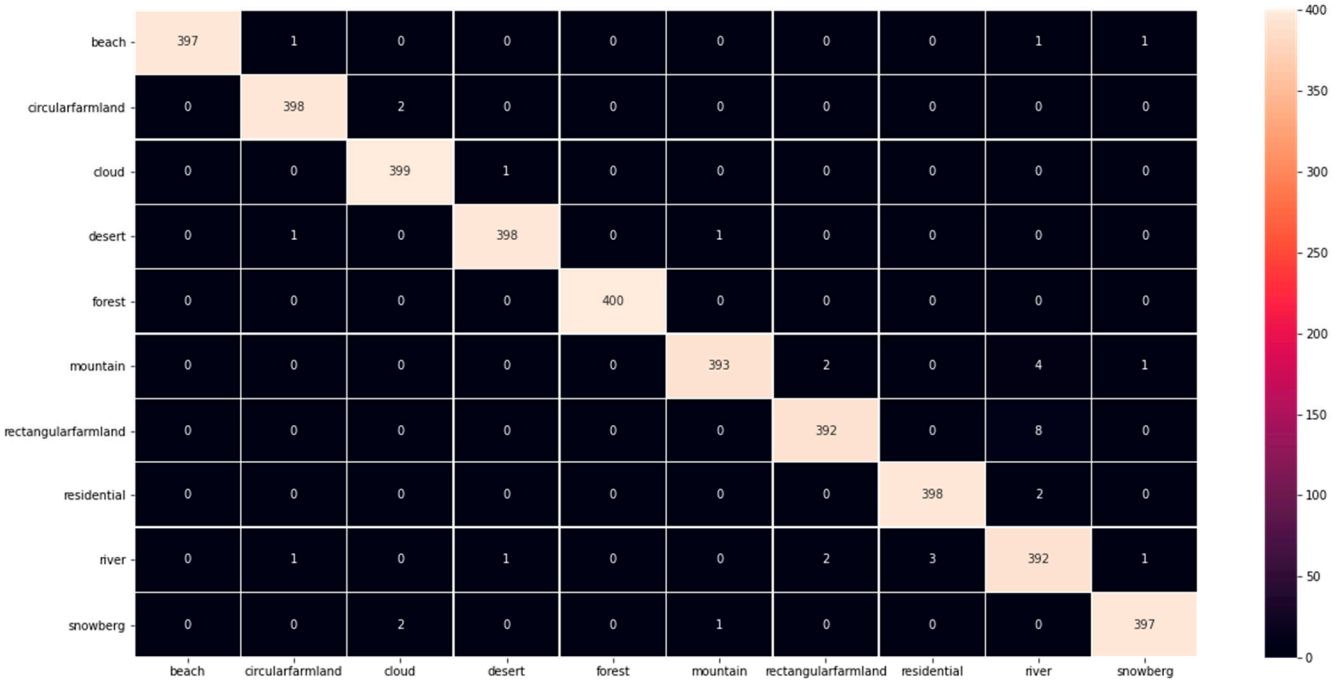

Figure 14. Confusion matrix for fine-tuning last 3 layers of ResNet-50.

Overall, fine-tuning the last three layers of the model resulted in an outstanding performance, surpassing many benchmarks. The results obtained using this technique were very close to the prior technique in which the entire model was fine-tuned. This is because these models were pre-trained on a very large dataset (i.e., ImageNet). Although the source domain was different to our applied domain and it contained natural images of regular objects, these models captured all the abstract and complex features in its feature extraction layers. We utilized these learned features for the remote-sensing domain by freezing those layers during the training. We fine-tuned only the final layers which captured the high-level features of the remote-sensing images. Since the low-level features are similar for all types of images, the deployed models were adapted easily to the NaSC-TG2 dataset for natural scene classification.

4.3. From Scratch Training on NaSC-TG2

In this case, we trained all layers of the ResNet-50 model on the NaSC-TG2 dataset from scratch using random weights as the starting point. Figures 15 and 16 show the graphs for ResNet-50 and VGG-19, respectively, displaying the accuracy curves for training and validation. As expected, these models performed poorly when compared to previous techniques in which transfer learning was used. Without the presence of pre-trained weights as the starting point, random weights were initialized. Contrary to previous experiments, ResNet-50 showed more fluctuations than VGG-19 during the training process. The obvious reason for this behavior is that it has a deeper structure than VGG-19; however, these models slowly converged, and ResNet-50 yielded a better accuracy than VGG-19 on the test dataset.

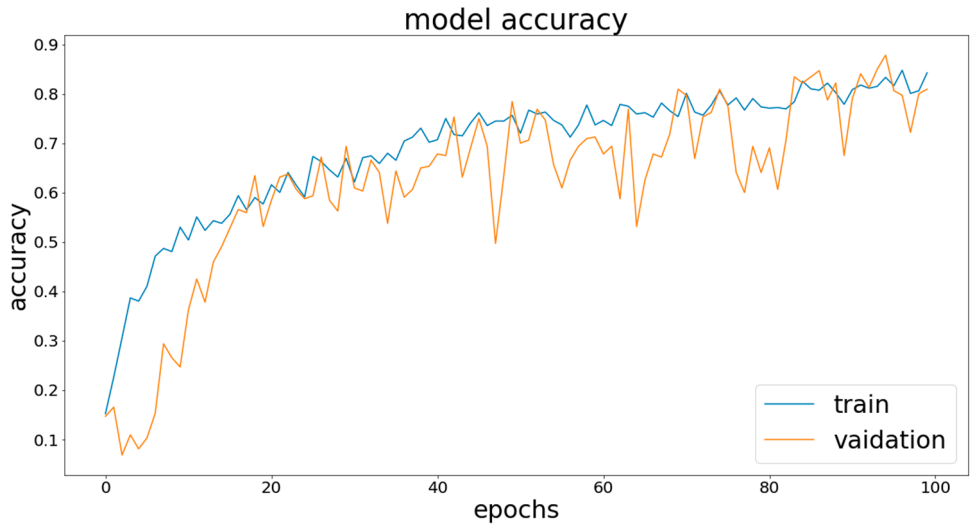

Figure 15. Training vs. validation accuracy curve during training of ResNet-50 from scratch.

Figure 16. Training vs. validation accuracy curve during training of VGG-19 from scratch.

After the training and evaluation, we achieved an overall accuracy of 82.67% and 76.72% for ResNet-50 and VGG-19, respectively. Still, the ResNet-50 model performed better than the VGG-19 model but these models were far behind transfer-learning-based methods in terms of accuracy. Detailed class-wise results for the ResNet-50 model training (from scratch) are presented in Table 3 for further analysis. Looking at the table, the river class showed the worst performance in terms of the precision (60.21%), recall (43.50%), accuracy (91.48%), and F1 score (50.51%). The forest class showed the best performance in terms of the precision (97.63%), accuracy (99.02%), and F1 score (94.00%) while the beach class showed the best result in terms of recall (94.00%).

Analyzing the confusion matrix of the ResNet-50 model training (from scratch) in Figure 17, we can clearly observe that the model yielded the lowest accuracy for the river class as many instances of the river class were confused as rectangular farmland, mountain, residential, or cloud, etc. The model also performed worse for snowberg as images for this class were confused with those of cloud, river, circular farmland, or others. The highest true positives were recorded for the forest class (TP: 370) and then for the desert class (TP: 368) out of 400 instances for each class in the test dataset. The highest false negatives were recorded for the river class (FN: 226) and then the snowberg class (FN: 114). The highest

false positives were recorded for the rectangular farmland class (FP: 175), for the cloud class (FP: 119) and then, for the mountain class (FP: 99).

Table 3. Results for training of ResNet-50 on NaSC-TG2 from scratch.

Class	Recall	Precision	Accuracy	F1 Score
Beach	0.9400	0.8931	0.9828	0.9160
Circular farmland	0.9050	0.8538	0.9750	0.8786
Cloud	0.8475	0.7402	0.9550	0.7902
Desert	0.9200	0.9608	0.9882	0.9400
Forest	0.9250	0.9763	0.9902	0.9499
Mountain	0.855	0.7755	0.9608	0.8133
Rectangular farmland	0.8675	0.6648	0.9430	0.7527
Residential	0.8700	0.9039	0.9778	0.8866
River	0.4350	0.6021	0.9148	0.5051
Snowberg	0.7150	0.9597	0.9685	0.8195
Overall	82.80%	83.30%	82.80%	82.52%

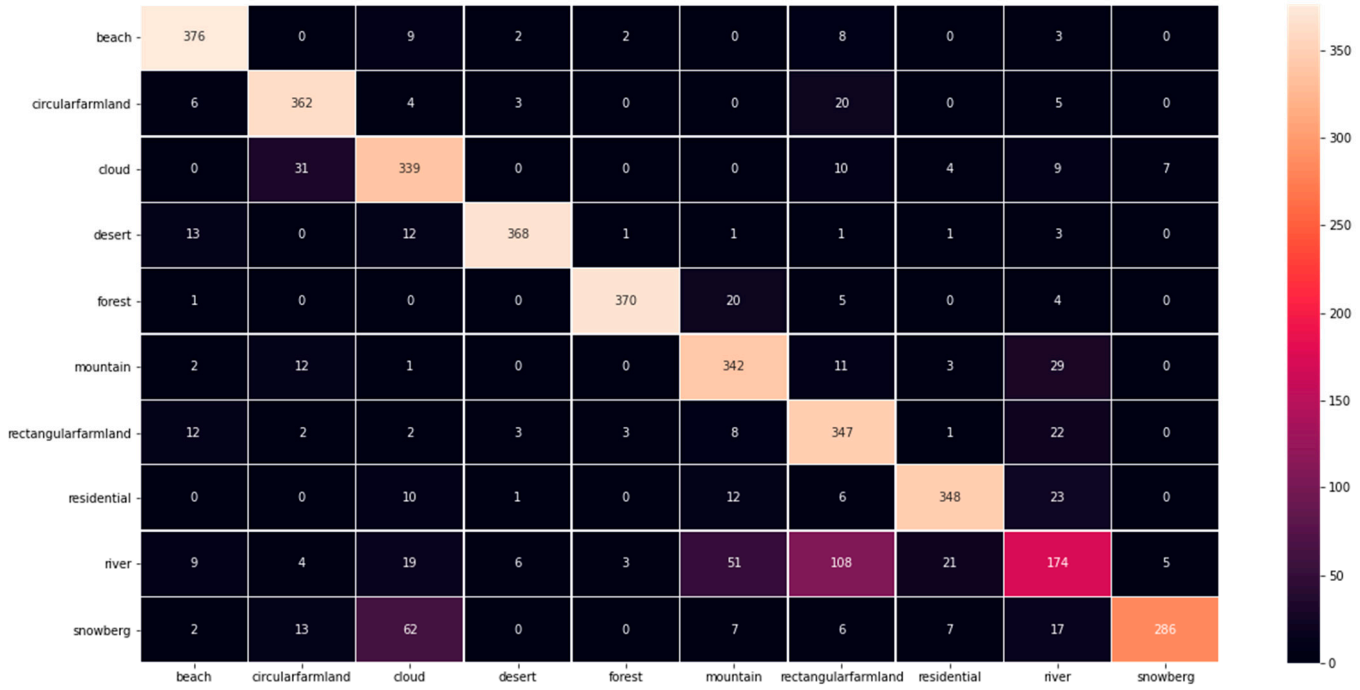

Figure 17. Confusion matrix for training of ResNet-50 from scratch.

4.4. Summary of Results

ResNet50's residual connections contribute to its enhanced generalization ability. The skip connections help in the flow of information, enabling the model to retain relevant information from earlier layers, even in deeper layers. This allows ResNet50 to capture both low-level and high-level features, leading to an improved performance in remote-sensing image classification tasks. VGG19, on the other hand, relies solely on sequential convolutional layers, which may not capture information as effectively across different scales. A comparison of the results for both ResNet-50 and VGG-19 are summarized in Table 4.

Table 4. Comparison of ResNet-50 and VGG-19 performance on NaSC-TG2.

Training Mode	Epochs	Overall Accuracy	
		ResNet-50	VGG-19
From scratch training	100	82.67%	76.72%
Last 3 layers fine-tuned	100	99.10%	97.28%
Entire model fine-tuned	100	99.50%	98.02%

On the other hand, VGG19 has fewer parameters compared to ResNet50, because it is a lighter model with fewer layers (see Tables 5 and 6). VGG-19 also tends to be faster than ResNe-50 in terms of inference time. This is primarily due to its smaller number of parameters and the architectural design that facilitates efficient information flow. With fewer parameters to process and compute, VGG-19 requires less computational resources during inference, resulting in faster predictions. Therefore, when considering the number of parameters and inference time, VGG-19 has an advantage over ResNet50. It offers parameter efficiency and faster inference, which can be beneficial in scenarios where computational resources or real-time processing are important factors.

Table 5. Training parameters and inference time for ResNet-50.

Training Mode	Trainable Parameters	Total Parameters	Inference Time
From scratch training	23,555,082	23,587,712	4.65 s
Last 3 layers fine-tuned	20,490	23,587,712	4.78 s
Entire model fine-tuned	23,555,082	23,587,712	4.59 s

Table 6. Training parameters and inference time for VGG-19.

Training Mode	Trainable Parameters	Total Parameters	Inference Time
From scratch training	20,029,514	20,029,514	3.08 s
Last 3 layers fine-tuned	5130	20,029,514	3.13 s
Entire model fine-tuned	20,029,514	20,029,514	3.23 s

Transfer learning leverages the knowledge gained from pre-training on a large-scale dataset and applies it to the target task. Since ResNet50 has a deeper architecture, its increased depth allows it to capture more complex and abstract features from the ImageNet dataset during its pre-training process. Therefore, when its knowledge is transferred to the remote-sensing domain, it performs better in the natural scene classification of remote-sensing images (See Table 4). ResNet50 employs skip connections that enable the model to propagate gradients more effectively during training, addressing the vanishing gradient problem. These skip connections help in the better flow of information and enable the model to learn more discriminative features. This allows the model to learn residual mappings, focusing on the differences between the input and output features. These residual connections help in the efficient propagation of gradients and enable the model to learn more easily. Here are some published results and their comparison with our model. (See Table 7). Hence, our state-of-the-art cross-domain transfer-learning-based model shows a superior performance when compared to these previously proposed methods for natural scene classification.

Table 7. Comparison of published results with our state-of-the-art model.

Methods	Precision (%)	Recall (%)	Accuracy (%)	F1 Score (%)
VGG-SA [28]	98.57	98.57	-	98.57
MobileNetV3-small pretrained [29]	-	-	99.08	-
PyHENet [30]	-	-	95.05	-
AlexNet [7]	-	-	89.39	-
VGG-16 [7]	-	-	89.59	-
GoogleNet [7]	-	-	87.76	-
ResNet-34 [7]	-	-	88.37	-
Inception v3 [7]	-	-	86.75	-
ResNet-50 (Ours)	99.50	99.50	99.50	99.50

To conclude, the state-of-the-art ResNet-50 model deployed for cross-domain transfer learning achieved the highest accuracy compared to other methods in the literature, indicating that it was mostly able to correctly classify images belonging to each class while avoiding false positives and false negatives. Overall, our approach of fine-tuning a pre-trained ResNet-50 model and VGG-19 model on the NaSC-TG2 dataset resulted in a significant improvement in the classification accuracy. This illustrates the usefulness of transfer learning across different domains, resulting in a better deep-learning performance on challenging natural scenes datasets.

4.5. Parameters Affecting the Performance of the Deployed ResNet-50

Figures 9, 10, 12, 13, 15 and 16 show that the performance of the deployed ResNet-50 model increased with the number of epochs but up to a certain threshold and then it became saturated. This is true for all three models of ResNet-50 and these are a standard procedure in the training. Likewise, we noted the similar behavior of the number of fine-tuned weights, i.e., the performance of ResNet-50 increased if we fine-tuned more layers but we noted that the choice of the last three layers was appropriate since, after it, the performance became saturated. As for hyper and other parameters of learning (for example, learning rate, number of epochs, optimizer, train/test split, batch size, image input size) are concerned, we fixed them for all experiments for a fair comparison.

5. Conclusions

This paper focuses on NaSC-TG2 dataset remote-sensing imaging for classification into ten different scenes facing multiple challenges. The main hazard is the non-availability of the scene texture and thus, the designing of handcrafted features is difficult to design. Therefore, deep learning is considered for automated and deep feature extraction and classification. Unfortunately, deep-learning structures require a number of weights to be optimized, which requires large datasets. Considering the non-availability of large datasets, transfer learning is considered. Unfortunately, the non-availability of an in-domain pre-trained model poses another challenge. Therefore, this study explored a cross-domain pre-trained model and in this regard, the ResNet-50 pre-trained on ImageNet was selected and fine-tuned on the NaSC-TG2 dataset. For this, we explored three different modes of transfer learning, i.e., fine-tuning the whole structure, fine-tuning only the last three layers, and from scratch. The experiments showed that the model from scratch was not performing well. On the other hand, the other two models performed almost the same (99.50% and 99.10% accuracies, respectively). It is because the initial layers extracted the basic gradient features, which are the same in these cross-domains, and thus, their final tuning is not affected significantly. Thus, we recommend cross-domain transfer learning for scene classification.

Author Contributions: Conceptualization, M.A. (Muhammad Akhtar), I.M., M.A. (Muhammad Adnan) and A.S.; Methodology, I.M., M.A. (Muhammad Adnan) and A.S.; Software, M.A. (Muhammad Akhtar); Validation, M.A. (Muhammad Akhtar); Formal analysis, M.A. (Muhammad Akhtar) and I.M.; Investigation, I.M. and A.S.; Resources, I.M.; Data curation, M.A. (Muhammad Akhtar); Writing—original draft, M.A. (Muhammad Akhtar); Writing—review & editing, I.M., M.A. (Muhammad Adnan) and A.S.; Supervision, M.A. (Muhammad Adnan). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The APC was covered by UiT The Arctic University of Norway.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tombe, R.; Viriri, S. Remote Sensing Image Scene Classification: Advances and Open Challenges. *Geomatics* **2023**, *3*, 137–155. [CrossRef]
2. Li, Y.; Zhang, H.; Xue, X.; Jiang, Y.; Shen, Q. Deep learning for remote sensing image classification: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1264. [CrossRef]
3. Dimitrovski, I.; Kitanovski, I.; Kocev, D.; Simidjievski, N. Current trends in deep learning for Earth Observation: An open-source benchmark arena for image classification. *ISPRS J. Photogramm. Remote Sens.* **2023**, *197*, 18–35. [CrossRef]
4. Cheng, G.; Xie, X.; Han, J.; Guo, L.; Xia, G.-S. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3735–3756. [CrossRef]
5. Kaul, A.; Kumari, M. A literature review on remote sensing scene categorization based on convolutional neural networks. *Int. J. Remote Sens.* **2023**, *44*, 2611–2642. [CrossRef]
6. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
7. Zhou, Z.; Li, S.; Wu, W.; Guo, W.; Li, X.; Xia, G.; Zhao, Z. NaSC-TG2: Natural Scene Classification With Tiangong-2 Remotely Sensed Imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 3228–3242. [CrossRef]
8. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
9. Mondal, A.; Kundu, S.; Chandniha, S.K.; Shukla, R.; Mishra, P. Comparison of support vector machine and maximum likelihood classification technique using satellite imagery. *Int. J. Remote Sens. GIS* **2012**, *1*, 116–123.
10. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
11. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [CrossRef]
12. Xia, G.-S.; Hu, J.; Hu, F.; Shi, B.; Bai, X.; Zhong, Y.; Zhang, L.; Lu, X. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
13. Zhou, Q.; Zheng, B.; Zhu, W.; Latecki, L.J. Multi-scale context for scene labeling via flexible segmentation graph. *Pattern Recognit.* **2016**, *59*, 312–324. [CrossRef]
14. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
16. Bu, S.; Han, P.; Liu, Z.; Han, J. Scene parsing using inference embedded deep networks. *Pattern Recognit.* **2016**, *59*, 188–198. [CrossRef]
17. Pohlen, T.; Hermans, A.; Mathias, M.; Leibe, B. Full-resolution residual networks for semantic segmentation in street scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4151–4160.
18. Tombe, R.; Viriri, S. Adaptive deep co-occurrence feature learning based on classifier-fusion for remote sensing scene classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *14*, 155–164. [CrossRef]
19. Boualleg, Y.; Farah, M.; Farah, I.R. Remote sensing scene classification using convolutional features and deep forest classifier. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1944–1948. [CrossRef]
20. Zhu, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Generative adversarial networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5046–5063. [CrossRef]
21. Fang, B.; Li, Y.; Zhang, H.; Chan, J.C.-W. Semi-supervised deep learning classification for hyperspectral image based on dual-strategy sample selection. *Remote Sens.* **2018**, *10*, 574. [CrossRef]

22. Xu, K.; Deng, P.; Huang, H. Vision transformer: An excellent teacher for guiding small networks in remote sensing image scene classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5618715. [CrossRef]
23. Zhao, G.; Zhang, Y.; Tan, J.; Li, C.; Ren, Y. A data fusion modeling framework for retrieval of land surface temperature from Landsat-8 and MODIS Data. *Sensors* **2020**, *20*, 4337. [CrossRef] [PubMed]
24. Li, E.; Xia, J.; Du, P.; Lin, C.; Samat, A. Integrating multilayer features of convolutional neural networks for remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5653–5665. [CrossRef]
25. Dai, X.; Xia, M.; Weng, L.; Hu, K.; Lin, H.; Qian, M. Multi-Scale Location Attention Network for Building and Water Segmentation of Remote Sensing Image. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5609519. [CrossRef]
26. Ghadi, Y.Y.; Rafique, A.A.; Al Shloul, T.; Alsuhibany, S.A.; Jalal, A.; Park, J. Robust object categorization and Scene classification over remote sensing images via features fusion and fully convolutional network. *Remote Sens.* **2022**, *14*, 1550. [CrossRef]
27. Koonce, B. (Ed.) ResNet 50. In *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*; Apress: Berkeley, CA, USA, 2021; pp. 63–72. [CrossRef]
28. Liu, Z.; Dong, A.; Yu, J.; Han, Y.; Zhou, Y.; Zhao, K. Scene classification for remote sensing images with self-attention augmented CNN. *IET Image Process.* **2022**, *16*, 3085–3096. [CrossRef]
29. Liu, H.; Qu, Y.; Zhang, L. Multispectral Scene Classification via Cross-Modal Knowledge Distillation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5409912. [CrossRef]
30. Chen, Q.; Wu, Y.; Wang, X.; Jiang, Z.L.; Zhang, W.; Liu, Y.; Alazab, M. A Generic Cryptographic Deep-Learning Inference Platform for Remote Sensing Scenes. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 3309–3321. [CrossRef]