# Remote Sensing Image Classification for Land Cover Mapping in Developing Countries: A Novel Deep Learning Approach

**Nzurumike Obianuju Lynda[1], Nwojo Agwu Nnanna [2††], and Moussa Mahamat Boukar [3††]**
nzurumikeuju@yahoo.com ,nagwu@nileuniversity.edu.ng ,musa.muhammad@nileuniversity.edu.ng
*Department of Computer Science, Nile University of Nigeria*

**Abstract**
Convolutional Neural networks (CNNs) are a category of deep learning networks that have proven very effective in computer vision tasks such as image classification. Notwithstanding, not much has been seen in its use for remote sensing image classification in developing countries. This is majorly due to the scarcity of training data. Recently, transfer learning technique has successfully been used to develop state-of-the art models for remote sensing (RS) image classification tasks using training and testing data from well-known RS data repositories. However, the ability of such model to classify RS test data from a different dataset has not been sufficiently investigated. In this paper, we propose a deep CNN model that can classify RS test data from a dataset different from the training dataset. To achieve our objective, we first, re-trained a ResNet-50 model using EuroSAT, a large-scale RS dataset to develop a base model then we integrated Augmentation and Ensemble learning to improve its generalization ability. We further experimented on the ability of this model to classify a novel dataset (Nig_Images). The final classification results shows that our model achieves a 96% and 80% accuracy on EuroSAT and Nig_Images test data respectively. Adequate knowledge and usage of this framework is expected to encourage research and the usage of deep CNNs for land cover mapping in cases of lack of training data as obtainable in developing countries.

***Keywords:*** *Convolutional neural network, remote sensing image classification, Land cover mapping, medium-resolution.*

## 1. Introduction

Land cover comprises the biophysical features covering the surface of the earth; this includes bare soils, forests, water bodies, rocks, vegetation, built-up etc. Land cover mapping identifies these different features, its location and proportion in the earth's land surface. The information is essential for many important applications including urban development, natural disaster monitoring, population estimation, geology, ecological management, agricultural purposes etc. However, the rate of population growth, poverty, socio-economic activities, and other environmental forces in developing countries has led to the exploitation, degradation and subsequent altering of these land cover features, leading to observable changes in the land cover[1]. The effect of this land-cover change is complex with significant consequences on humans, its environment and the economy[2]. It is therefore very crucial to have information not only on existing land cover but also the capability to monitor the dynamics of land cover change in real time. This will help to develop sustainable policies to mitigate the impacts of these challenges, facilitate sustainable management of the land resources as well as meet the increasing demands for basic human needs and welfare in developing countries.

The advent of remote sensing technology: which is the science of detecting and monitoring the physical characteristics of an area by measuring its reflected and emitted radiation at a distance gave birth to space-based satellites. These satellites acquire remote sensing images at different spatial resolutions- low, medium, high and very high, which provides researchers and scientists with a cost-effective means of mapping the land cover. Its proliferation in the past decades has provided a basis for quantifying rates of land cover change around the world. It allows large-scale surveys to be conducted within a short time with repeated surveys possible at interval of less than 24hours[3]. However, the problem is in interpreting the content of a remote sensing imagery otherwise known as satellites image classification, since remote sensing s sensors do not label or interpret the content of its images.

In the past, numerous techniques such as manual interpretation and tradition machine learning techniques have been developed and used to classify these different resolution images. While the initial technique depends on visual and analytical capability as well as the observer's knowledge to extract useful information from a remote sensing image. The later uses computational and statistical methods based on the spectral features reflected by the different physical features on the earth surface. Notable disadvantages of these methods include low accuracy, huge training time, difficulty in understanding the structure of algorithm and inability to generalize well for a large- scale learning problem [4]. Additionally, there is often extreme variability of spectral reflectance associated with various land cover types. This variability poses major challenges in mapping and analyzing land cover types based solely on their spectral properties

Manuscript received February 5, 2022
Manuscript revised December 20, 2020
https://doi.org/10.22937/IJCSNS.2020.22.2.28

Deep learning (DL) techniques in particular convolutional neural networks (CNN) have recently emerged as a powerful solution to solve computer vison tasks such as image classification due to its ability at solving complex tasks and its representational power at identifying features[5]. It is a specialized form of a neural network model. In CNNs, a number of convolutional and pooling layers are designed to learn the set of features present in an image. They are based on an end-to-end learning process, from raw data such as image pixels to semantic labels.

For a CNN model to automatically classify an image into different features present in it. It must be trained or programmed to recognize the features using data known as 'training data'. The model attempts to learn the visual features present in the training images associated with each label, and classify unlabeled or unseen images known as 'test data' accordingly. However, it needs massive amounts of data to achieve satisfactory model accuracy. Such amount of data is difficult to obtain in many everyday scenarios [6]. With limited dataset, deep CNN's work perfectly on the training data but do not generalize well to test data resulting in poor performance [7]

Due to the need for sufficient datasets for training a CNN model, several benchmark, hand- labelled datasets were introduced in developed countries by different groups to enable machine-learning based research. Examples of these datasets include (MNIST, CIFAR-10, ImageNet) consisting of natural images, (UC Merced (UCM) Aerial Image Dataset (AID), EuroSAT) consisting of remote scene images, (CheXpert, MIMIC-III, fastMRI) for medical images etc. Unlike the natural images dataset which can range up to millions of samples [8] remote scene image datasets are usually very few in number. Acquiring high resolution remote sensing image data is very expensive and time-consuming. While, freely accessible medium resolution RS images can be downloaded from earth observation programs such as European space Agency's (ESA) and the United States National Aeronautics and space Administration's (NASA), it is very expensive in terms of expertise, money and time especially for developing countries to label the right quantity of RS image patches needed to train a high performing CNN model.

Recently, the use of transfer learning (TL) has facilitated the use of these huge natural image datasets to other scientific fields that have less available data such as health care, remote sensing, airplane detection etc. [9]. It exploits knowledge present in labeled training data from a source domain to enhance an already trained (Pre-trained) model's performance on a target domain, with limited training data [10].

The pre-trained models are re-trained on these RS training datasets to allow the model learn the features present in the data. The new model or learned weights can then be used to classify the test data from the same dataset. However, applying the learned model directly to RS test data from different dataset leads to poor accuracy. Example is the use of models trained with EuroSAT [10] dataset produced from satellite images of European cities to classify satellite images of African countries, acquired using different scenarios from those of the training data This is because unlike natural images there is high variability between RS data of different locations. Recent research has also shown that using varying image resolution hinders classification performance of CNN based image classifiers [11]. Also, external factors such as resolution, lighting, background, sensor types, view angles, and post-processing can cause the feature space divergence of the same task in 2 different datasets [12]. Additionally, these models are trained to solve specific tasks and has to be re-bult from scratch once the feature-space distribution changes. This problem has posed a serious challenge to the usage of these CNN models directly for classifying RS data in training data scare environments as obtainable in developing countries. There is need to develop robust CNN models with stronger generalization capability.

Inspired by the above, we propose a new approach for improving the generalization ability of CNN models for remotely sense image classification task in developing countries.

Specifically, the main contribution of this paper is to:

1) Propose a deep CNN framework that can classify RS test data from a dataset different from the training dataset
2) To experiment on the ability of the model in 1 to classify pixels in a novel medium resolution remote sensing image data (Nig_Images) into five major land cover classes (Bare land, built-up, water-bodies, farm land and forest)

The reminder of this paper is arranged as follows, Section 2 presents review of related work; section 3 discussed the methodology of our proposed work; section 4 presents the experimental results and discussions and finally section 5 provides the conclusion.

## 2. Literature Review

Several studies have been carried out in the area of deep learning particularly convolutional neural network (CNN). Different stakeholders have successfully used it for image classification tasks in areas such as health industry, face recognition, robotics, computer vision etc. However, this review is more concerned with newer works that employed convolutional neural network for the classification of remotely sensed images.

Remote sensing (RS) is the major source of spatial information related to the earth's surface, offering a wide range of sensors and platforms to monitor land cover and its spatial distribution [13]. With the development in earth observation technologies, the acquired RS images have increased drastically and how to effectively mine these massive volumes of remote sensing data are now the new challenge [14].

The revolution in data science and the growth of artificial Intelligence gave birth to Deep learning. Deep learning (DL) models in particular convolutional neural networks have emerged as a powerful solution to approach many computer vision tasks including image classification. It has become one of the most widely accepted emerging technology due to its ability at solving complex tasks and its representational power at identifying features. The history of DL dates back to the mid 90's with the creation of a computer model based on the neural networks of the human brain. However, it did not get much attention until a breakthrough in deep learning research in the mid-2000s. It provides a new approach for analyzing remote sensing data. It has offered a compelling alternative to traditional classification techniques and has the ability to handle the growing earth observation data. Recently, there has been a rapid surge of interest in its use within the remote sensing community.

D. Lu & Q. [15] discussed the major steps in image classification using DL. They noted that designing a suitable classification system (DL architecture /model) is a prerequisite for a successful classification task. For a CNN model to automatically classify an image into different features present in the image. It must be trained or programmed to recognize the features using a large number of similar features known as 'training data'. Getting vast amounts of labeled RS data for training a model can be really difficult, considering the money, time and effort it takes to label data points. This has posed a huge challenge to the usage of Convolutional Neural Networks (CNN) in remote sensing image classification tasks.

The introduction of transfer learning (TL) technique allows training CNN models on a base dataset for a specific task and then using those learned features/ model parameters as the initial weights in a new classification task. A state-of-the-art model is trained on large scale images e.g., ImageNet and the weight of the pre-trained network is then fine-tuned another dataset. By transfer learning, researchers are able to take advantage of the expensive resources (expertise, training data and computational power) that were used to acquire it. The recent focus of research for RS image classification is on the use of TL.

R. P. De Lima & K. Marfurt [16] investigated the performance of transfer learning from CNNs pre−trained on natural images for remote−sensing scene classification versus CNNs trained from scratch only on the remote sensing scene classification dataset themselves. They evaluated different depths of two popular CNN models—VGG 19 and Inception V3 using three different sized remote sensing datasets. The results show that transfer learning from models trained on larger more generic natural images outperformed transfer learning from models trained on smaller remotely sensed dataset, they concluded that learning can be transferred from one domain to another and therefore will provide a powerful tool for remote-sensing scene classification.

For TL to be successfully used, the training samples and testing samples must be in the same domain or feature spaces, such that the input feature space and data distribution characteristics are the same[10]. However, this is not true for transfer learning from natural objects to RS data otherwise known as heterogenous transfer learning. The feature spaces between the natural objects and RS data are non-equivalent and are generally non-overlapping.

Generalization performance (how good a model is at learning from a given data and applying the learnt information in a different domain) is known to reduces as the source and target domain become less similar leading to overfitting and subsequently poor performance of the model.

Many approaches have been developed to solve the problem of heterogenous transfer learning.

## 3. Methodology

The goal of this research is to propose a deep CNN framework that can classify RS test dataset different from the dataset used in the model training and to experiment on the ability of the model to classify pixels in a medium resolution RS image data into five major land cover classes (Bare land, built-up. Water-bodies, farm land and Forest). To do this, we first import a pre-trained model (ResNet-50) using transfer learning (TL) technique, we fine- tuned the model and retrained it using EuroSAT, a large-scale remote sensing image dataset. We further integrated augmentation and ensemble learning techniques to the model to create a unique model able to generalize to a novel dataset. We describe the data we used in Sect 3.1, the model framework design in 3.2, and the model implementation details in 3.3.

## 3.1. Datasets

The approaches aim at improving the generalization ability of the model. M. Xie et al. [17] used a sequence of transfer learning steps to design a novel ML approach. Zhong Chen et al. [9] utilized TL for Airplane detection in remote sensing images. They used VGG16, a pre-trained network as the base network and replaced the fully connected layer with a secondary network structure and finally fine-tuned with the limited number of airplane available training samples

Some researches combined other popular techniques with TL. Grant. J. Scott et al [59] investigated the use of Deep CNN for land-cover classification in high-resolution remote sensing imagery. To overcome the lack of massive labelled data sets, they utilized two techniques in conjunction with DCNN: TL with fine-tuning and data augmentation tailored specifically for remote sensing imagery.

While developed countries are successfully using TL technique, the absolute lack of RS training data repositories in developing countries presents a huge challenge. Due to the diverse distributions of objects, spectral shifts caused by different acquisition conditions of images and other external factors, deep models trained on a certain set of annotated RS images may not be directly used to classify images acquired by different sensors or from different geo-locations [18]. There is need to develop robust models with stronger generalized capability for remote scene classification in data scarce areas e.g., developing countries.

Two sets of datasets were used in our study

1) EuroSAT dataset: an RS data repository for benchmarking land cover classification tasks was used. Introduced by Patrick Helber et al [10], EuroSAT are large-scale Sentinel-2, labelled and geo- referenced remote sensing image patches measuring 64 ×64 pixel. It was gotten from 34 countries in Europe. EuroSAT dataset originally consists of 10 land use and land cover (LULC) classes with each class between 2000 and 3000 and a total of 27,000 divided into 10 generic land cover types (Highway, Industrial, Residential, Pasture, Forest, Herbaceous Vegetation, Sea, Lake, River, Permanent Crop and Annual Crop). We reclassified similar land cover classes from the EuroSAT dataset to suit our land cover classes of interest (bare soil, built- up, farmland forests, water bodies) as shown in Table 1.

2)Nig_Images Dataset: To evaluate and test the ability of our model to classify out of domain images correctly, we generated a novel dataset (image patches) from Sentinel-2 satellite image of Abuja city in Nigeria. downloaded via their website: https://scihub.copernicus.eu/dhus/#/homeSentinel-2 images. The image was further processed and enhanced. A total of 25 image patches, five for each land cover class were extracted for testing the model. We called the dataset Nig_Images dataset. The full methodology is described in [19].

**Table 1**. Reclassified EuroSAT landcover classes

| **land cover used in EuroSAT** | **New re-classified Land Cover Classes** | **Class Label** | **Total number after reclassification** |
|---|---|---|---|
| Highway | Bare soil | 1 | 2500 |
| Residential and industrial | Built-up | 2 | 5500 |
| Annual crop, herbaceous vegetation, pasture, permanent crop | Farm Land | 3 | 10500 |
| Forest | Forest | 4 | 3000 |
| Rivers, lakes, Sea | Water Bodies | 5 | 5500 |

## 3.2. Model Framework Design

### 3.2.1 Transfer Learning

First, we developed our base model using Transfer learning (TL) technique. The following steps were taken:

a)Select the pre- trained model: The first step while using TL technique is to import a pretrained model of our choice. Pretrained models have different characteristics which we considered when making our choice. The most important are network accuracy, speed, and size. We also considered models that have reported reasonably high accuracy, low training error and can be trained without difficulty. Also, considering the complexity of our data, the depth of the network is considered as it helps extract high level features. For our pre- trained model, we selected a ResNet-50 model. ResNet50 has been trained on a subset of the ImageNet database (http://www.image-net.org) and can classify images into 1000 object categories (e.g., ball, birds, shoes and cup). See [23] for more details on the structure and workings of a ResNet-50 model.

b)Fine tune the ResNet-50 model: A typical CNN model for image classification is usually composed of 2 parts: the convolution base which is composed of a stack of convolutional and pooling layers and the classifier or top model usually composed of one or more staked fully connected (fc) layers. The main purpose of the classifier is to classify the image based on the detected features. The goal of fine-tuning the ResNet- 50 model is to reshape the last layer (classification layer) of the top model to have the same number of inputs as before, but have the same number of outputs as the number of land classes in the reclassified EuroSAT dataset. It allows us to reuse the filters learned during the previous training. Note that a ResNet-50 has 177 layers in total, corresponding to a 50-layer residual network. We froze the weights of the initial layers (1 to 174) so they are not updated as the new model is trained and the fine- tuned the last 3 layers. Freezing the weights of these layers will significantly speed up our network training and prevent over- fitting to our new remote sensing dataset.

c)Additional layers: Using the ResNet-50 model as the base network, we then added three additional layers:

**Max-Pooling Layer**: A -pooling layer is added to summarize the activations from the last convolution layer of the pre-trained model. Pooling also provides an approach to down-sample feature maps by summarizing the presence of features. This technique preserves relevant features and removes irrelevant ones. We used a max-pooling layer instead of the average pooling layer (AVG) at the end of the original ResNet-50 network as seen in [20]]. Max pooling layer allows us to extract the important features in our remote sensing data. AVG from literature is known to smooth out an image and prevents the sharp features from been identified. This layer is then flattened using the Flatten() function to convert the Matrix to single array.

**Two fully connected (densely) layers** of sizes 1024 and 5, respectively, with 'relu' as the activation function were added. The final fully connected (dense) layer reduces the input to the number of classes using a softmax activation function.

**Dropout layer**: A dropout of size 0.5 (50%) is added in between the 2 fully connected layers. This is a common value used in most literature. Dropout technique is a regularization method which forces randomly selected neurons to be "dropped-out" with a given probability during training. The network is forced to learn several independent representations of the data. This helps the network to generalize better, increase accuracy and thereby prevent the model from overfitting

The architecture of the proposed base model is shown in figure 1

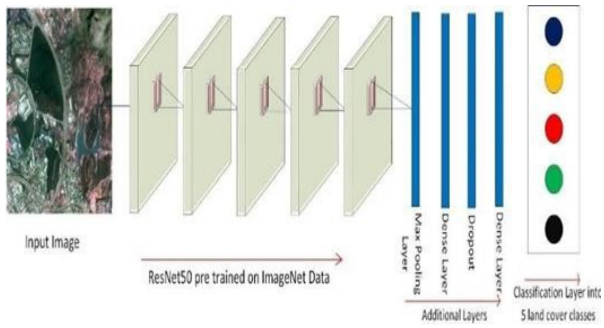


Figure 1. Architecture of the proposed model

### 3.2.2) Training the CNN model

We trained our model on the reclassified EuroSAT train dataset using the model.fit() function, Our input consists of a set N image patches of size 3×64 × 64, each labelled with one of the land cover classes. So, we have 3× 64 ×64 matrices corresponding to the RGB pixel intensity values of the image. The training phase is when the network "learns" from the data it will be fed with. We used ImageDataGenerator available in Keras to read images in batches directly from the train folder. Using augmentation process, we generated three different sets of training dataset. Our aim is to use the different datasets to produce different model weight. First, we randomly applied traditional transformations such as: rescaling, rotations, flipping zoom etc. and saved the model weight as train_datagen1. Note that the augmentation processes are derived from the Keras and TensorFlow libraries. Then, we re-trained the initial model weight using two other different subsets of datasets train_datagen2 and train_datagen3 obtained using different augmentation techniques. Train_datagen2 utilized photometric transformations such as color space: brightness and contrast randomization. The TensorFlow library provides more advanced capabilities to modify the color space. Different lighting conditions will be considered in the transformation to simulate images taken under high-contrast or low-visibility conditions. The RGB values was manipulated with simple matrix operations to increase and decrease the contrast and brightness of the image. While train_datagen3 utilized random erasing, a data augmentation technique discussed in [21]. Remote sensing images are often occluded or covered by clouds. Random Erasing solves the problem of occlusion which is a limiting factor to the generalization ability of CNN's. Introduced by Z. Zhong [21], it masks different portions of the image with rectangular region of arbitrary size during training. Precisely, it works by randomly selecting an n × m patch of an image within a mini-batch and masking it with random values or mean pixel value thus, generating training images with various levels of occlusion. It forces the model to learn more descriptive features about an image. It complements the other augmentation approaches to further improve the performance accuracy. It is easily implemented with ImageDataGenerator in Keras using get_random_eraser(). Note that 10% of each of the augmented datasets were reserved as the test set and named (test_ datagen1, test_datagen2 and test_datagen3) respectively.

3.2.3) Ensemble learning: Finally, we combined the generated weights to produce a single-robust model using ensemble learning technique. This is the process by which multiple models, are strategically generated using subset of the base dataset generated above. It combines several individual models to obtain better generalization performance [24]. Ensembles are designed to increase the accuracy of a single classifier by training several different models and combining their predictions to output a single prediction.

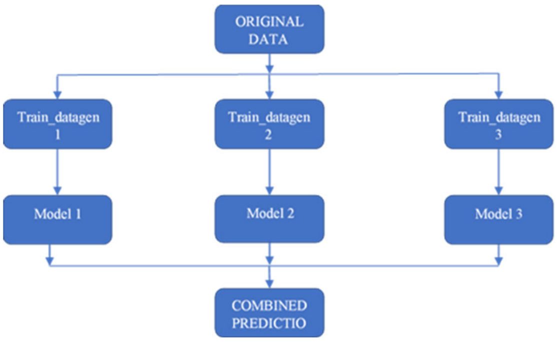


Figure 2. illustration of an ensemble process

### 3.3) Model Implementation Details

Our model implementation was done on Keras. Keras is a powerful and easy to use API for developing and evaluating deep learning models. These libraries provide a complete deep learning toolkit for training and testing our models. This phase consists of three major procedures as described below:

3.3.1)Load the designed model: Keras provides access to the ResNet-50 model available via the Applications API, and has functions to load the model with the pre-trained weights. When loading the model, we set the “include_top” argument to False, which allows us to add and train the new additional layers. The Sequential model API allows us to add the new layers. We also specified the “input_tensor” argument which allows the expected fixed-sized input size of the ResNet-50 model (224 × 224 × 3) to be changed to (64× 64× 3).

3.3.2)Compile the model: At this step, we configured the learning process by specifying the parameters and hyperparameters as shown in table 3.2. We used random hyperparameters and also experience from other literature. Table 2 details the best choice for our hyperparameters.

Table 2. Hyperparameters used

| S/N | Hyper-parameter | value |
|---|---|---|
| 1 | Optimizer | RMSprop gradient descent |
| 2 | Batch size/Epoch | 32/10 |
| 3 | Activation function | RELU |
| 4 | Loss function | Cross Entropy |
| 5 | Learning rate | 0.0001 |

3.3.3)Validation Phase: Our validation dataset is used to give an estimate of our model’s performance while fine- tuning the model’s hyper parameters. After each epoch, the accuracy is computed on the validation set by plotting the performance on the train and validation dataset over each epoch for the three datasets A, B and C to provide learning curves and insight into how well the models are working while tuning the hyper parameters. Validation curves also give an indicator if training longer improves the model’s performance. The models for each set that reaches the lowest value of the lost function (at training time) or error rate is chosen and is used in the test phase.

3.3.4)Testing Phase: The testing phase is where what the network has learned is evaluated. First the individual models were tested on the EuroSAT data and Nig_Images test dataset.

We then loaded the combined model and tested on the Nig_Images test dataset to predict the classes that make up the image patches (Built- ups, Forest, Vegetation, Water bodies, Bare soil).

The methodology for designing and implementing our proposed CNN model is shown in figure 3.

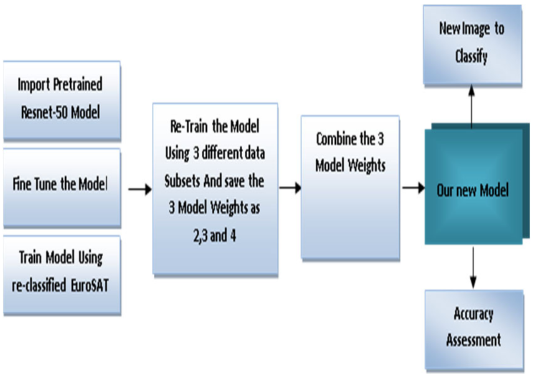


Figure 3: Methodology for our proposed model

## 4. RESULTS AND DISCUSSIONS

### 4.1 Results

This section analyzes the results obtained in our study. We have fine-tuned the pre-trained ResNet-50 model which consists of 48 convolutional layers and added additional layers comprising of 1 max pooling layer, 2 densely connected layer and a drop-out layer. The model summary is given in table 3.

Table 4a. Classification result for Experiment1

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bare-land | 0.96 | 0.94 | 0.95 | 1051 |
| Built-Up | 0.97 | 0.99 | 0.98 | 1049 |
| Farm-land | 0.97 | 0.98 | 0.97 | 1050 |
| Forest | 0.95 | 0.96 | 0.96 | 1051 |
| Water-Bodies | 0.95 | 0.93 | 0.94 | 1052 |
| | | | | |
| Accuracy | | | 0.96 | 5253 |
| macro avg | 0.96 | 0.96 | 0.96 | 5253 |
| weighted avg | 0.96 | 0.96 | 0.96 | 5253 |

Table 4d. Classification result for Experiment4

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bare soil | 0.80 | 0.80 | 0.80 | 5 |
| Built-up | 0.80 | 0.80 | 0.80 | 5 |
| Farmland | 0.80 | 0.80 | 0.80 | 5 |
| Forest | 1.00 | 1.00 | 1.00 | 5 |
| Water Bodies | 0.60 | 0.60 | 0.60 | 5 |
| Accuracy | | | 0.80 | 25 |
| macro avg | 0.80 | 0.80 | 0.80 | 25 |
| weighted avg | 0.80 | 0.80 | 0.80 | 25 |

Table 4b. Classification result for Experiment2

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Bare soil | 0.00 | 0.00 | 0.00 | 5 |
| Built Up | 0.40 | 0.80 | 0.53 | 5 |
| Farmland | 0.29 | 0.40 | 0.33 | 5 |
| Forest | 0.71 | 1.00 | 0.83 | 5 |
| Water Bodies | 1.00 | 0.20 | 0.33 | 5 |
| | | | | |
| Accuracy | | | 0.48 | 25 |
| Macro avg | 0.48 | 0.48 | 0.41 | 25 |
| Weighted avg | 0.48 | 0.48 | 0.41 | 25 |

Table 4c. Classification result for Experiment3

| Classification report: | | | | |
|---|---|---|---|---|
| | precision | recall | f1-score | support |
| Bare soil | 1.00 | 0.90 | 0.95 | 250 |
| Built-up | 0.97 | 1.00 | 0.98 | 550 |
| Farmland | 0.98 | 0.98 | 0.98 | 1050 |
| Forest | 0.97 | 0.98 | 0.98 | 300 |
| Water Bodies | 0.97 | 0.97 | 0.97 | 550 |
| Accuracy | | | 0.98 | 2700 |
| macro avg | 0.98 | 0.97 | 0.97 | 2700 |
| | | | | |
| Weighted avg. | 0.98 | 0.98 | 0.98 | 2700 |

### 4.2 Discussion

The lack of training data in developing countries is a major obstacle to harnessing the potentials of DL in remote sensing image classification tasks such as land cover mapping.

From our research work we observed the following:

1) That deep networks trained for image recognition in one task (ImageNet) can be efficiently transferred and used for RS classification tasks. This was demonstrated on the EuroSAT dataset, where above 96% mean accuracy was achieved using a ResNet-50 model pretrained on ImageNet. This backs the findings obtained by previous researches

2) We proved the high variability which occurs among remote sensing images of different locations which was demonstrated when the model trained with EuroSAT data was initially tested on Nig_Images data yields a drop from 96% to 46% in accuracy. We postulated that the drop was as a result of the variability between remote sensing images of different locations.

3) Augmentation and Ensemble learning techniques have the ability to produce training data more robust to achieving a high generalization ability. From geometrical (physical) to photometric and random erasing all provided high accuracy on the classification task. Since DL models require a large amount of data to obtain satisfactory results in remote sensing image classification tasks. These techniques offer promising solution to tackle data scarcity challenge experienced in developing countries.

Our model was able to effectively classify Nig_Images dataset which are RS data different from the data which was used for training the initial model.

## 5. CONCLUSION

The lack of training data in developing countries is a major obstacle to harnessing the potentials of DL in remote sensing image classification tasks such as land cover mapping. In this paper, we designed and implemented a novel technique based on deep convolutional neural network for remote sensing image classification in data scare environments. Our model was able to classify a novel dataset: Nig_Images dataset. We observed a performance accuracy of 80%. The model was based on the integration of three important techniques: transfer learning, augmentation and ensemble learning. In contrast to existing remote sensing image classification research works which focuses on finding solutions for improving (deep) learning algorithms in case of limited training data, we proposed a multi- phase deep learning approach method which works in cases of non-availability training data. This research work contributes to the use of the state-of-the-art deep learning technique for land cover mapping in developing countries.

## References

[1] S. M. Herrmann, M. Brandt, K. Rasmussen, and R. Fensholt, “Accelerating land cover change in West Africa over four decades as population pressure increased,” *Commun. Earth Environ.*, vol. 1, no. 1, pp. 1–10, 2020.

[2] F. Oluwabunmi, A. Joan, A. Alaga, and A. Debora, “Geospatial Analysis of Landuse and Landcover Dynamics In,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 1, no. 2, pp. 149–157, 2016.

[3] S. Basu, S. Ganguly, and S. Mukhopadhyay, “DeepSat - A Learning framework for Satellite Imagery DeepSat – A Learning framework for Satellite Imagery,” no. September, 2015.

[4] A. Sharma, X. Liu, X. Yang, and D. Shi, “A patch- based convolutional neural network for remote sensing image classification,” *Neural Networks*, vol. 95, pp. 19–28, 2017.

[5] H. Song, Y. Kim, and Y. Kim, “A Patch-Based Light Convolutional Neural Network for Land- Cover Mapping Using Landsat-8 Images,” pp. 1– 19, 2019.

[6] F. Ponzio, G. Urgese, E. Ficarra, and S. Di Cataldo, “Dealing with lack of training data for convolutional neural networks: The case of digital pathology,” *Electron.*, vol. 8, no. 3, 2019.

[7] L. Alzubaidi *et al.*, *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*, vol. 8, no. 1. Springer International Publishing, 2021.

[8] R. P. De Lima and K. Marfurt, “Convolutional Neural Network for Remote - Sensing Scene Classification: Transfer Learning Analysis,” 2020.

[9] Z. Chen, T. Zhang, and C. Ouyang, “End-to-end airplane detection using transfer learning in remote sensing images,” *Remote Sens.*, vol. 10, no. 1, pp. 1–15, 2018.

[10] O. Day and T. M. Khoshgoftaar, “A survey on heterogeneous transfer learning,” *J. Big Data*, vol. 4, no. 1, 2017.

[11] S. P. Kannojia and G. Jaiswal, “Effects of Varying Resolution on Performance of CNN based Image Classification An Experimental Study,” *Int. J. Comput. Sci. Eng.*, vol. 6, no. 9, pp. 451–456, 2018.

[12] J. Zhang, W. Li, P. Ogunbona, and D. Xu, “Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective,” *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–35, 2019.

[13] M. Pashaei and H. Kamangir, “Review and Evaluation of Deep Learning Architectures for Efficient Land Cover Mapping with UAS Hyper- Spatial Imagery: A Case Study Over a Wetland,” 2020.

[14] J. Song, S. Gao, Y. Zhu, and C. Ma, “A survey of remote sensing image classification based on CNNs,” *Big Earth Data*, vol. 3, no. 3, pp. 232–254, 2019.

[15] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," vol. 1161, 2007.

[16] R. P. De Lima, A. Bonar, D. D. Coronado, K. Marfurt, and C. Nicholson, "Deep convolutional neural networks as a geological image classification tool," no. Dl.

[17] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, "Transfer Learning from Deep Features for Remote Sensing and Poverty Mapping," 2014.