Combining Convolutional Neural Networks and SVM for Land Use and Land Cover Scene Classification

MSc Research Project
Data Analytics

Preety Kumari
Student ID: x18128556

School of Computing
National College of Ireland

Supervisor: Prof. Noel Cosgrave

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Preety Kumari
Student ID:	x18128556
Programme:	Data Analytics
Year:	2020
Module:	MSc Research Project
Supervisor:	Prof. Noel Cosgrave
Submission Due Date:	12/12/2019
Project Title:	Combining Convolutional Neural Networks and SVM for Land Use and Land Cover Scene Classification
Word Count:	6145
Page Count:	17

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	28th January 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Contents

1 Introduction **1**

2 Related Work **2**

2.1 Data Source 2
2.2 Deep Learning Models for Image Applications 3
2.3 Hybrid CNN Model for Image Applications 4

3 Research Methodology **6**

3.1 Data Collection 6
3.2 Data Preparation and Pre-processing 6
3.3 Modeling 7
3.3.1 Deep Feature Extraction 7
3.3.2 Multi-class Classification 8
3.4 Model Evaluation 10

4 Design Specification and Implementation **10**

4.1 Bottleneck Feature Extraction 11
4.2 Multi-class Classification 11
4.2.1 Fully-connected layers 11
4.2.2 Support vector machines 12

5 Evaluation **13**

5.1 Classification without hyper-parameter tuning 13
5.2 Classification with hyper-parameter tuning 13
5.3 Results and Discussion 13

6 Conclusion and Future Work **14**

7 Acknowledgement **15**

Combining Convolutional Neural Networks and SVM for Land Use and Land Cover Scene Classification

Preety Kumari
x18128556

Abstract

Growth of population over the years have led to intensive changes in our landscape. This has led to changes in the environment and hence is very important to monitor landscape changes constantly and accurately. Land use and land cover (LULC) classification is a method for analysing these changes which could be very helpful for urban land use planning. This study proposes a model containing a classifier on top of deep feature extraction block for land use and land cover scene classification. A benchmark dataset, NWPU-RESISC45 for scene classification is used to gather 16 LULC scene classes for this study. ResNet-152 was used to extract deep features of dimension 2048 from deep layers. Support vector machine (ResNet-152-SVM) and stack of fully-connected layers (ResNet-152-FC), used for classification in this research were evaluated using the extracted bottleneck features. Both models were trained using some initial set of parameters after which hyper-parameter tuning was applied on these models. The performance of the classifiers was found to have improved after hyper-parameter tuning. The results showed that ResNet-152-FC with an overall classification accuracy of 94.91% outperforms the ResNet-152-SVM model that had an overall accuracy of 93.45% for LULC scene classification.

Keywords: Land Use and Land Cover Scene Classification; Image Classification; Residual Neural Networks; Support Vector Machines; Fully-Connected Layers

1 Introduction

With the growth of population in the last few decades, it has become very important to understand the changes in our landscape. Land use and land cover classification is very helpful in determining changes in the landscape. In the past years, a lot of research has been done in this domain. Land cover mainly describes the landscape change in terms of vegetation and physical land surface type. Land use on the other hand describes the changes in landscape due to human activities such as residential areas, grassland, etc. Researchers spatially map and categorize these changes into land use and land cover classes. Image classification is carried out by firstly extracting deep features from images and then performing final classification. Residual neural networks when used for a variety of image applications such as object detection Son et al. (2019) and image classification Yang, Rottensteiner and Heipke (2018), have shown better results than other deep learning models such as VGGNet-16, AlexNet, Inception modules. Since image classification is a two-step process, separation of these two blocks is quite easy. Guo et al. (2019) proposed a combined architecture of residual neural network and support vector machines for deep feature extraction and image classification respectively which showed great results. SVM has also been used in various classification problems and given good results Liu et al. (2007).

In this study, a combined model consisting of a feature extractor block and classifier block is implemented for LULC scene classification. A pre-trained ResNet with 152 layers is used to extract bottleneck features of dimension 2048 from deep layers. These features are then be fed to a classifier for final classification outcome. Classifiers are first trained on a set of data and then evaluated on the test set. Two classifiers are implemented in this research. First is a stack of fully-connected layers activated by softmax function and second is support vector machines. Hyper-parameter tuning is performed on both the classifiers using random search algorithm with K-fold cross validation. Both the classifiers were evaluated with (tuned model) and without hyper-parameter (base model) tuning.

This paper consists of the following sections: in Section 2 critical evaluation of previous state-of-the-art methodologies using past research works is done, Section 3 describes the details of the overall research methodology, Section 4 describes the complete design and implementation of the research methodology, in Section 5 describes the various evaluation methods that were used in this research and finally Section 6 contains summary and future work for this research.

Research Question: Can a combined model of residual neural networks and support vector machines outperform state-of-the-art model containing stack of fully-connected layers for land use and land cover scene classification?

Research Objectives:

1. To assess the performance of support vector machine classifier (ResNet-152-SVM) when compared to a stack fully-connected layers classifier (ResNet-152-FC) implemented for LULC scene classification.

2. To assess the performance of support vector machine (ResNet-152-SVM) classifier when compared to its state-of-the-art performance for waveform recognition.

2 Related Work

2.1 Data Source

Land use land cover classification is a very important area of research and many datasets have been proposed in the past years that are described below:

1. **UC Merced**: Yang and Newsam (2010) introduced UC Merced Land-Use dataset for scene classification. It contained 21 classes with 100 images per class of each measuring 256 x 256 pixels. Large images for various regions of the work were collected from USGS National Map Urban Area Imagery and patches of size 256 x 256 pixels were manually extracted from these images to form the final dataset.

2. **EuroSAT**: First proposed by Helber et al. (2019) for land use and land cover classification. Sentinel 2 satellite images covering 13 spectral bands was converted

into patches measuring 64 x 64 pixels. This dataset consisted of 10 classes with 2000-3000 images per class measuring 64 x 64 pixels.

3. **NWPU-RESISC45**: Cheng et al. (2017) proposed a large-scale benchmark dataset for REmote Sensing Image Scene Classification (RESISC). This dataset consisted of a total of 31,500 remote sensing images belonging to 45 scene classes covering more than 100 countries and regions all over the world. Each class included 700 images with a size of 256 x 256 pixels belonging to RGB (Red, Green and Blue) color space. The spatial resolution varied from about 30 m to 0.2 m per pixel.

All these databases were studied in depth and many limitations were found in UC Merced and EuroSAT datasets such as small number of images per class or small-scale of scene classes which is not sufficient for training of deep learning models. RESISC45 on the other hand is a large-scale dataset with rich image variations such as spatial resolution, translation, object pose, background, appearance, occlusion, illumination and view point that could be very useful in accurate training of deep learning models.

2.2 Deep Learning Models for Image Applications

Machine learning techniques are used in a variety of applications such as object detection, image segmentation and image classification. Machine learning models have widely been used in the past for image classification and have shown promising results. One such study was presented by Liu et al. (2007) that uses support vector machine (SVM) for multi-class classification problem. Existing multi-class SVM methods such as one-against-one and one-against-all was compared with an improved technique of one-against-one method. Although the improved one-against-one reduced the model parameters but one-against-one method showed the least number of misclassified samples. Similar study was presented by Gidudu et al. (2007) that compared one-against-one and one-against-all methods for SVM multi-class problem but both methods showed insignificant difference in kappa value and number of misclassified samples. The included SVMs were linear, polynomial, quadratic and radial basis function (RBF).

Machine learning models such as decision trees, support vector machines, K nearest neighbour, artificial neural networks (ANN), convolutional neural networks (CNN) and deeper learning models of CNN such as VGGNet, GoogLeNet have been widely used for image classification problems. Yang, Rottensteiner and Heipke (2018) adopted CNN-based methodologies namely, SegNet for land use and LiteNet for land cover classification. These were trained from scratch, fine-tuned and also ensembles model of these two were evaluated and found that ensembles model outperformed others. Also, ResNet architecture of CNN model has shown great results when used for different applications such as image recognition and object detection He et al. (2015). They proposed deep residual networks (ResNets) with multiple layers and compared them with state-of-the-art approaches such as VGG-16, GoogLeNet. ResNet-152 model performed the best for ImageNet [1] dataset as compared to other methods with least top-1 (21.43%) and top-5 (5.71%) error rate. For object detection, ResNet-101 outperformed other models.

The deep residual networks have shown impressive results on image classification problems such as scene classification Yang, Kim and Kim (2018), clinical image classification Han et al. (2018) and LULC classification Helber et al. (2019). An ensembles

[1] ImageNet: `http://www.image-net.org/`

method combining CNNs was used by Ju et al. (2017) for image classification. Ensembles of ResNet models (ResNet-8 and ResNet-110) were trained multiple times and trained with different checkpoints. The classification results showed that residual neural networks combined with other networks can show great results. Inspired by deep residual networks, Yang, Kim and Kim (2018) applied pre-trained ResNet-152 model with and without squeeze-and-excitation for scene classification problem. In this, SE-ResNet-152 had the best accuracy when used without any data augmentation. Similarly, Helber et al. (2019) evaluated multiple models such as BoVW using SIFT features and trained SVM, shallow CNN (2 layers), GoogLeNet, ResNet-50 in which Resnet-50 showed the best classification accuracy. Although residual networks have shown promising results for some image classification problems but its performance for other applications always needs to evaluated. Deep learning models such as VGG16, DenseNets-121, Inception (v-1/2/3/4) and ResNet (50/101/152) were evaluated by Too et al. (2019) for plant disease image classification and Jannesari et al. (2018) for breast cancer image classification. These models were fine-tuned for application specific softmax layer and parameters. Even though ResNet showed great results for breast cancer image classification. It failed to outperform DenseNets-121 for plant disease classification.

2.3 Hybrid CNN Model for Image Applications

A standard deep learning technique for classification problems uses softmax layer at the top. But there are exceptions, particularly in papers Tang (2013) Elleuch et al. (2016) Wolfshaar et al. (2015) Copur et al. (2018), where support vector machine is used by replacing the last softmax layer which is responsible for classification. Originally proposed for binary classification, SVMs have shown excellent results in multi-class classification problems too. There are many ways to replace softmax layer with SVM. One such way was used in handwritten digit recognition, scene classification and facial expression recognition by Tang (2013) and gender recognition by Wolfshaar et al. (2015). They used linear SVM with L2-regularized hinge loss in the last layer of the model. The main difference between softmax and linear SVM here was the loss function. On one hand, linear SVM performed better than softmax for facial expression recognition and scene classification. But failed to outperform softmax was handwritten digit recognition and gender recognition. Another method proposed by Elleuch et al. (2016) for Arabic handwritten recognition had SVM with RBF kernel in the last layer of CNN which showed best accuracy as compared to CNN-based model. Similar approach was adopted by Copur et al. (2018) for vehicle detection using aerial imagery that used soft-margin classifier SVM in the last layer. They also used histogram of oriented gradients (HOG) features with SVM. Comparison showed that CNN+SVM outperformed the other two approaches, namely CNN-based model and HOG+SVM.

Another method of combining CNN with other classifiers proposed by Cheng et al. (2017) for scene classification problem used deep learning-based CNN as bottleneck feature extractor and SVM for classification of these bottleneck features. Multiple kinds of features such as unsupervised feature learning models, handcrafted features, CNN-based features and fine-tuned CNN based features were evaluated for feature extraction. Deep CNN-based features and fine-tuned deep CNN-based features outperformed previous state-of-the-art models. For CNN-based approaches namely, AlexNet and VGGNet-16 features were extracted from the second last fully-connected layer and in case of GoogLeNet features were extracted from the last pooling layer. Among these VGGNet-

16 and fine-tuned VGGNet-16 had the best classification accuracy when compared other approaches that were used. Similar effort was made by Nogueira et al. (2016) for scene classification, that compared pre-trained, fine-tuned pre-trained and trained from scratch ConvNets such as OverFeat, AlexNet, VGG-16, GoogLeNet, CaffeNet and PatreoNet as feature extractors that are fed to linear SVM for classification. Almost for all variants of fine-tuned pre-trained ConvNets when used as feature extractors showed good results for different kinds of datasets. Use of SVM by replacing the last softmax layer of almost every ConvNet was a better solution for all scenarios.

Deep CNN models are composed of huge number of layers that can be categorized into 5 layers namely, input layer, convolutional layers each followed by pooling layers, few fully-connected layers and one output layer. The output layer which is composed of a softmax function is the one responsible for classification and specific to a problem. Hence this layer can be easily replaced with other classifiers. Convolutional neural network for bottleneck feature extraction in combination with SVM for classification has been used for multiple image classification tasks Qi et al. (2017) Nijhawan et al. (2017) and have shown great results in the past. OverFeat CNN network combined with a linear SVM using one-against-all and one-against-one strategy have excelled in multiple image classification problems such as scene classes, bird species, flower categories classification and object detection Razavian et al. (2014). Similarly, a combination of n-dimensional CNNs and SVM with RBF kernel was evaluated for sentiment analysis by Cao et al. (2015). All these proposed architectures, used CNN as deep feature extractor that returns feature vector of some dimension and SVM was used a separate unit for classification purpose. Deep features extracted by CNN was used as training and testing data for SVM.

Object detection is another domain where this hybrid model of CNN for feature extraction and a separate classifier for classification have shown good results in the past Vakalopoulou et al. (2015) Son et al. (2019). The choice of CNN layer for feature extraction is separate in different scenarios. Both Vakalopoulou et al. (2015) and Son et al. (2019) used second last fully-connected layer of CNN for feature extraction which returned vectors of size 4096. But for building detection Vakalopoulou et al. (2015) used SVM as classifier whereas, Son et al. (2019) used Faster R-CNN for construction workers detection. Both these approaches showed better results as compared to their baseline models. Feature based transfer learning approach was proposed by Guo et al. (2019) that used pre-trained residual neural network ResNet-152 and inception network Inception-v3 as feature extractor for radar signals classification. For both of these models, the last layer was removed and the rest of the network was used without any changes for feature extraction. A vector of size 1 x 1 x 1024 and 1 x 1 x 2048 was returned by ResNet-152 and Inception-v3 respectively. Further these features were fed to SVM for classification. ResNet-152-SVM outperformed Inception-v3-SVM network with better overall classification accuracy.

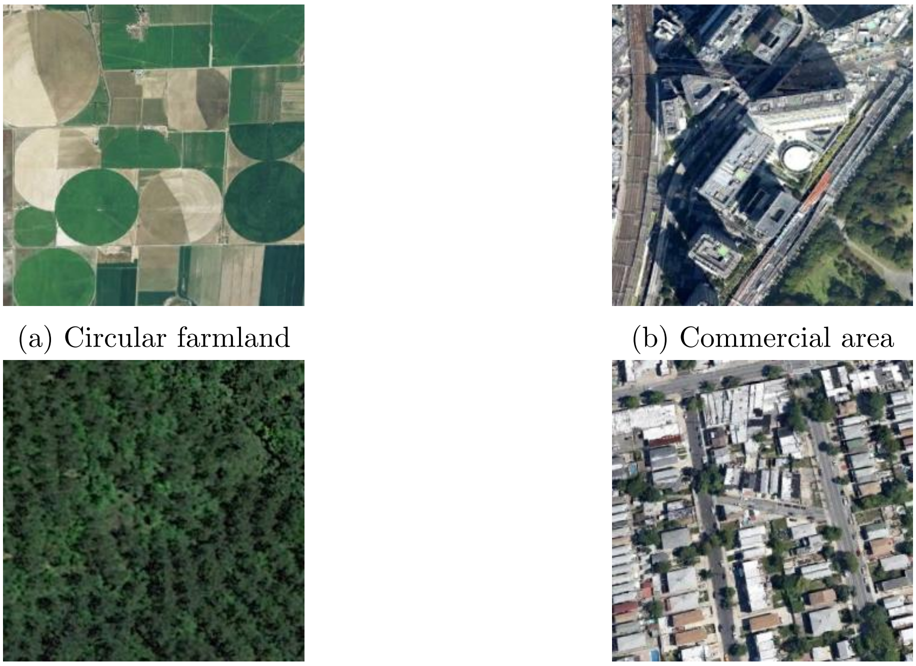
In this research, a combined model of deep residual networks for feature extraction block with SVM for land-use and land-cover scene classification similar to Guo et al. (2019) is implemented. SVM will be compared against a stack of fully-connected layers that is fed by features extracted from deep residual network for classification.

3 Research Methodology

This section explains the detailed methodology for implementing this research work. CRISP-DM approach is followed for developing this project.

3.1 Data Collection

Multiple databases such as UC Merced land use, EuroSAT and NWPU-RESISC45 were examined. Demerits such as small scene classes and insufficient data per class was found in the first two datasets. On the other hand, RESISC45 was found to be a large-scale dataset with rich variations in images and hence was chosen to be used in our research. This dataset is an open-source dataset which is available for download from NWPU-RESISC45 website [2]. A manual selection of scenes from the RESISC45 dataset to determine land use and land cover classes was performed as suggested by Cheng et al. (2017). Out of 45 scene classes 16 scenes, namely, circular_farmland, commercial_area, dense_residential, desert, forest, industrial_area, lake, meadow, medium_residential, mountain, rectangular_farmland, river, sparse_residential, snowberg, terrace and wetland were selected to form our LULC scene dataset. Sample images of few of these classes are shown in Figure 1.



(a) Circular farmland

(b) Commercial area

(c) Forest

(d) Dense residential

Figure 1: Samples images from few land use and land cover classes.

3.2 Data Preparation and Pre-processing

After collection of dataset exploratory data analysis was performed on it. The final dataset consisted of 16 LULC classes with 700 images per class. Detailed description of the dataset which is used in our analysis is listed in Table 1.

[2] NWPU-RESISC45 dataset: `http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html`

Table 1: Raw dataset description.

Dataset	Images per class	Scene classes	Total images	Spatial resolution (in meters)	Image size
Land-Use and Land-Cover Scene	700	16	11,200	$\sim$ 30 to 0.2	256 x 256

Since the count of images per class is equal, no class imbalance was present. Raw images if applied to any classification problem does not produce reliable results. Hence pre-processing is an important step before applying any machine learning model. Data is first split into train and test set in the ratio of 70%-30%. Data augmentation such as rotation, zooming, horizontal flip and shear intensity was applied on the train dataset. Both train and test images were resized to 224 x 224 pixels which is the desired input size for the CNN model being used in this research. Pre-processing to an adequate format required by the model is also performed on the images. Detailed description of the final dataset after data-augmentation and pre-processing that will be used for further analysis is shown in Table 2.

Table 2: Description of final dataset.

Dataset	Images per class	Scene classes	Total images	Image size
Training Set	980	16	15,680	224 x 224
Testing Set	210	16	3,360	224 x 224

3.3 Modeling

The choice of appropriate machine learning techniques is very important for good results. LULC scene classification which is a multi-class classification problem will be evaluated in this research using supervised machine learning approach. A supervised image classification algorithm is based on selection of image samples representative of a specific class by the user that will be used for training of the classification model. Support vector machines and convolutional neural networks are few examples of supervised machine learning algorithm for classification problems that have been used in the past Gidudu et al. (2007) Cheng et al. (2017) Helber et al. (2019) have given promising results.

An architecture combining convolutional neural networks for feature extraction and support vector machines for classification will be used in this research. Previous studies Cao et al. (2015) Vakalopoulou et al. (2015) Guo et al. (2019) have successfully combined CNN and SVM for classification problems and have shown better results than other algorithms such as VGGNet, GoogLeNet, SVM. Figure 2 shows the block diagram of the overall architecture for the implemented framework.

3.3.1 Deep Feature Extraction

Deep learning approaches such as convolutional neural network is used for analysing visual imagery has shown great results in the past studies. CNNs are nothing but regularized versions of multilayer perceptrons (MLP) which means fully-connected network.

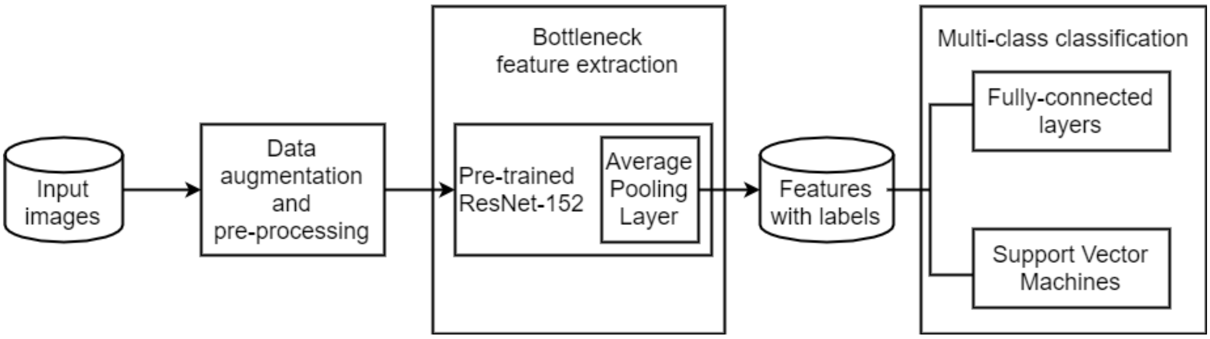


Figure 2: Overall design of the implemented model.

A convolutional neural network consists of an input layer, multiple hidden layers and an output layer. The hidden layers consist of a stack of convolutional layers having ReLU activation function and pooling layers, a series of fully-connected layers and normalization layers. The last layer of CNN is generally a softmax loss function which is responsible for predicting a class from multiple mutually exclusive classes.

Deep learning models failed to outperform their shallow counter-parts in multiple applications due to reasons such as optimization function, vanishing/exploding gradient problem and initialization of the network. To solve the above issues, ResNet block that won the first prize in ILSVRC 2015 for classification competition was introduced. It consists of a 'Skip Connection' identity mapping is used to add the output of previous layer to the next layer by matching the channels of the shortcut connection thus increasing the performance of deep learning models. Hu et al. (2017) introduced squeeze-and-excitation blocks to further improve the overall performance of ResNets. Residual blocks have shown good performance in multiple applications such as object detection Vakalopoulou et al. (2015) and image classification Yang, Kim and Kim (2018).

Inspired by deep residual networks great performance, ResNet-152 architecture of CNN which is pre-trained on ImageNet dataset for bottleneck feature extraction similar to Guo et al. (2019) is implemented in this research. They used pre-trained CNN model to extract bottleneck features of size 1 x 1 x 1024 from average pooling layer. For this research, pre-trained models will be used as training the model from scratch has many disadvantages including large data and computational power requirement. Also demonstrated by Razavian et al. (2014), deep hidden layer units can be used as generic image descriptors. The model used in this research is 152 layers deep. Global average pooling layer is selected as the feature extraction layer for this model which returns features of size 2048. This model requires number of samples and sample inputs of size (224 x 224 x 3) and outputs flattened features of size 2048. Here 224 x 224 is the height and width of the image samples and 3 refers to RGB (color space) values. Pre-processing is done on all input samples to convert them into a desired format that can be fed to ResNets as input. The extracted bottleneck features will be further used for training and testing of the classifiers.

3.3.2 Multi-class Classification

In machine learning domain, classification problem of multi-class data means classifying samples into more than two output classes. LULC scene classification is also a multi-class classification problem which will be evaluated in this research work. Krizhevsky et al. (2017) used three different classifiers namely, maximum entropy, naïve bayes and support vector machines. All of them were tested on combinations of different features. Stacking fully-connected layers of CNN with softmax activated layer can be used as classifier Krizhevsky et al. (2017). The softmax layer returns a probability distribution of a sample over each class. SVM is also used for classification of deep features extracted from convolutional neural network layers Razavian et al. (2014). SVM and fully-connected layers with softmax layer are used for classification in this research.

1. **Support Vector Machines**: SVM can be useful for both binary as well as multi-class classification problems Joachims (1998). The simplest way to extend SVM for multi-class classification problem is by using one-against-one and one-against-all approaches described in Gidudu et al. (2007). In this study, SVM with one-against-one strategy will be used. For a N-way multi-class problem N(N-1)/2 binary classifiers are trained. During training each of these classifiers receive samples from a pair of classes to learn to distinguish between these classes. For prediction a voting scheme is used by N(N-1)/2 classifiers. The class with highest number of +1 votes is predicted by the combined classifier. SVM contains multiple kernel functions such as quadratic, polynomial and gaussian radial basis function (RBF). SVM algorithms use different kernels for text, sequence or other kinds of data Cao et al. (2015) Elleuch et al. (2016). The most used SVM kernel is the RBF kernel because of its finite and localized response along the x-axis and also when their is no prior knowledge about the data. It comes with cost (C) and gamma parameters. C is the penalty applied on SVC for every misclassified sample. A small value of C means classifier is okay with misclassified outputs and with larger values of C, classifier is heavily penalized for misclassified data points. Gamma defines the decision boundary and thus decision region of the kernel. Low value of gamma means the 'curve' of decision boundary is low and hence the decision region of kernel is very broad. RBF kernel is the most widely used kernel in SVM for different classification tasks and have shown great results in the past Tang (2013) Elleuch et al. (2016). But linear kernels have also proven to be good at varieties of classification problems Copur et al. (2018) Guo et al. (2019) Vakalopoulou et al. (2015).

2. **Fully-connected Layers**: The top layers of any convolutional neural network consist of few dense layers and an activation function for classification. Mostly the activation function used in classification problems is softmax, but one should experiment before making any choices. Softmax layer returns a probability distribution over the classes. Similar to Krizhevsky et al. (2017) Zeiler and Fergus (2013), two fully connected layers and one output layer with softmax activation will be used as classifier in this study. Each of the fully-connected layer or dense layer is followed by dropout layer. Dropout layer is used in large networks for randomly dropping out nodes during training to reduce overfitting and improve network generalization capabilities. Another parameter known as optimizer is used for compiling the built model. Multiple optimizers are present out of which stochastic gradient descent will be used in this study. The choice of optimizer is based on past results. This optimizer has shown good results in the past when used with CNN, hence it has been chosen for this research work. All models are compiled with the chosen optimizer and loss function. Loss function is an important metric used for evaluating model performance. Weight parameters learned by the model are determined by

minimizing the chosen loss function. Cross-entropy loss is used to return probabilities between the range 0 to 1 for whether a sample belongs to a positive class or not.

Fine-tuning model by selecting an optimal set of parameters is very crucial for learning models. Both the implemented classifiers namely, fully-connected layers and SVM were fine-tuned using random search algorithm with cross validation approach. This validation approach starts by divides data into N number of folds. In the first iteration, first fold is used for testing and the rest of the folds are used for training. In this manner, every time a different set of data is fed to the model for training and testing. This is very important as the same model can behave differently for different sets of data.

3.4 Model Evaluation

Model accuracy is to measure the effectiveness of the model. A number of metrics such as accuracy, precision and recall are present for evaluating model performance. These metrics for a per class basis can be gathered from the classification report. All the metrics such as precision, recall and F1 scores of this report are calculated using true positive, false positive, true negative and false negative outcomes. Confusion matrix is another important report which is useful in determining the number of misclassified samples. It is a N x N matrix, where N is equal to the number of classes.

This study focuses on multi-class classification problem, hence metrics such as Rank-1 and Rank-2 accuracy, AUC ROC score are also very useful here. A model returns multinomial distribution of predicted class and these probabilities should sum up to 1. For Rank-1 accuracy we check if the top class (class with the highest probability) is same as the actual label. Similarly for Rank-5 accuracy we check if the actual label is one of the 5 predictions (top 5 classes with highest probabilities). Higher value means more number of correct predictions. AUC ROC score is calculated in which, ROC (Receiver Operating Characteristics) curve is the measure of how well the model can distinguish between two classes and AUC (Area Under the Curve) is the area under the ROC curve. Since this metric is used to distinguish two classes we have customised this to suite our multi-class classification problem.

Many metrics are present for evaluating multi-class classifiers. One such metric that works best for evaluating these classifiers is the Matthews Correlation Coefficient (MCC). It is used for evaluating multi-class classifiers by calculating the correlation coefficient between predicted and actual class. A value of +1 represents exact prediction, -1 represents inverse prediction and 0 means random prediction.

4 Design Specification and Implementation

In this study, a hybrid model for deep features extraction and classification is implemented for LULC scene classification. These classifiers are used on top of deep convolutional neural network to facilitate transfer learning. A pre-trained ResNet with 152 layers (ResNet-152) is selected for extracting bottleneck features from pre-processed images. These features are then fed to different classifiers that are responsible for classification based on the features extracted from deep residual neural network. Post data augmentation and pre-processing, the dataset generated contained 15,680 samples for training and

3,360 samples for testing belonging to 16 scene classes. No class imbalance was found as the number of samples present in both train and test set per class was same. After this the pre-processed input is sent for further processing.

4.1 Bottleneck Feature Extraction

Pre-trained ResNet-152 model was used for deep feature extraction in this research. Base ResNet-152 model is modified by removing the last dense layer of the model which was specific to the ImageNet classification problem on which this is pre-trained. The model used in our research takes an input dimension of 224x224 pixels with RGB color space, hence pre-processing was performed on raw input images to convert them into a format suitable for the used residual network. Last global average pooling layer output which is of dimension 1x2048 is extracted from ResNet-152 model. Both sets of data were fed to this block for bottleneck features extraction. Output of this block is features of dimension (n_samples, 2048) and labels of dimension (n_samples, n_classes) where n_samples refers to number of samples in the set and n_classes denotes the number of scene classes for classification. Post feature extraction the output of this layer is flatten and saved to avoid execution of this block again and again. For saving the bottleneck features, a Pythonic interface known as h5py package was used. HDF5 is a binary data format that facilitates storage of huge amount of data.

4.2 Multi-class Classification

Two kinds of classifiers are compared in this research work for LULC scene classification namely, support vector machines and fully-connected layers. A detailed description of design and implementation of each of these classifiers are given below:

4.2.1 Fully-connected layers

The first classifier that was used in this research is a stack of two fully-connected layers followed by one softmax function activated layer. Each of the two fully-connected layers were followed by a dropout layer equal to 0.5 to avoid overfitting. ResNet-152 expected an input vector of size 2048 and returned probabilities of samples belonging to a class. The output of first and second FC layer is a vector of size 1024 and 512 respectively. The size of last layer is specific to a given problem. Since the dataset used for this research contains 16 land use and land cover classes, the last layer of this model was built with 16 neurons i.e., 16 output nodes and activated by softmax loss function. This model was compiled using stochastic gradient descent (SGD) optimizer similar to the approach followed by Karpathy et al. (2014). Categorical cross-entropy loss is used to train CNN to output probabilities over all the classes for each input. Learning models require a set of hyper-parameters for execution. The hyper-parameters used in this model are learning rate, weight decay, momentum, batch size and epochs. Base model was trained with initial set of parameters, learning rate=0.01, decay=1e-5, dropout=0.5 and batch_size=28 for 30 epochs. Fine-tuned version of this model was also evaluated in this research work. Overall architecture of the fully-connected layers used as classifier in this research is shown in Figure 3.

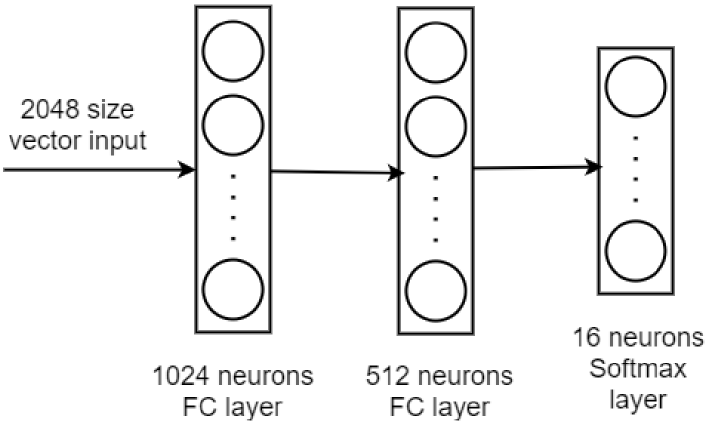


Figure 3: Design architecture of the fully-connected layers.

4.2.2 Support vector machines

Initially support vector machines were developed for binary classification problem but can easily be extended to multi-class classification problems Liu et al. (2007). In our research work, a support vector machine with multi-class support known as C-Support Vector Classification (SVC) was used. SVC can take parameter to return a one-vs-one decision function of libsvm [3] having shape (n_samples, n_classes * (n_classes -1)/2) or one-vs-all decision function of shape (n_samples, n_classes) which is similar to other classifiers. Here n_samples refers to the total number of samples and n_classes is same as number of classes used for classification. Both the approaches have show great results in the past Liu et al. (2007) Gidudu et al. (2007). However, for multi-class classification problem one-vs-one technique should be used and hence, it was implemented in this research work. SVM has another important parameter such as kernel, C and gamma. Kernel is nothing but a set of mathematical functions that are used to transform data, gamma is used to set the decision region of the kernel and C is used to apply penalty on the classifier for misclassified samples. Radial basis function (RBF) is the most commonly used kernel in SVMs. Initially RBF kernel with C=10 and gamma=1e-5 was used for classification, after which hyper-parameter tuning using K-fold cross validation approach was applied on it to search for optimal set of parameters of the model for better outcome. The model was finally trained with the optimal set of parameters determined using after cross validation. SVM with initial parameters mentioned before as well as fine-tuned version was evaluated in this research work.

For hyper-parameter tuning, the choice of search algorithm and number of iterations is application specific and experiments should be performed before finalizing them. Random search algorithm and grid search algorithm with 10-fold cross validation and 20 iterations were experimented for hyper-parameter tuning. The same set of optimal parameters were returned by both approaches. Finally random search was used in this study due to its faster execution.

[3] libsvm: https://en.wikipedia.org/wiki/LIBSVM

5 Evaluation

Multiple metrics are present for evaluating the performance of deep learning models and are previously discussed in the Section 3.4. In this section, two different classifiers namely, stack of full-connected layers and support vector machine will be evaluated using these metrics. Hyper-parameter tuning was also applied on these models to further improve the overall classification accuracy. The experimental setup of these two models with and without hyper-parameter tuning for scene classification is described further.

5.1 Classification without hyper-parameter tuning

In this experiment, both the models were trained using some initial values for hyper-parameters required by the model. This forms the base model without hyper-parameter optimization. In case of fully-connected layers classifier, the model was trained for 30 epochs using initial values for hyper-parameters, learning rate=0.01, weight decay=1e-5, dropout=0.5 and batch size=28. The optimizer used for this model was stochastic gradient descent (SGD) with momentum=0.9 as it was found to be an appropriate optimizer in previous researches. The second classifier, SVM was trained with initial hyper-parameters such a C=10, kernel='rbf' and gamma=1e-5. All the metrics described in Section 3.4 were calculated for both the models for evaluation.

5.2 Classification with hyper-parameter tuning

Fully-connected layers and SVM classifiers were hyper-tuned with 10-fold cross validation strategy using random search algorithm. After this, a model is created using the best parameters determined by hyper-parameter tuning. Finally training and evaluation is performed on this model. For SVM, hyper-parameters that were tuned are kernel type, gamma and C. In case of fully-connected layers, learning rate, weight decay, dropout, batch_size and epochs were hyper-tuned. All the metrics mentioned in Section 3.4 were calculated for further evaluation of both the models. Even though for multi-class classification generally, radial basis function or linear kernels are used in SVM Elleuch et al. (2016) Wolfshaar et al. (2015), but in this case, quadratic was found to be the best performing kernel after hyper-parameter tuning. Corresponding values of C and gamma were also selected after hyper-parameter tuning to create model with best parameters. This is similar to Gidudu et al. (2007), where it was proposed that all the kernel variants performed equally good when used for multi-class classification.

5.3 Results and Discussion

Both the classifier were trained on initial parameters as well as tuned hyper-parameters. Table 3 shows the classification results before and after applying hyper-parameter tuning for both the models. Model containing stack of fully-connected layers performed better than SVM in terms of metrics such as overall accuracy, precision, F1 score, recall, area under the ROC curve that are determined from the classification report. Rank-1 and rank-5 accuracy is also found to be better in case of fully-connected layers. But even though fully-connected layers performed better than the SVM model, hyper-parameter tuning made less than 1% of improvement in the overall performance of fully-connected layers as compared to 1∼2% improvement in case of SVM. Approximately, a percentage difference was observed in the rank-1 accuracy of both the models before and after applying hyper-parameter tuning, whereas there was a negligible difference in the rank-5 accuracy of these models. This shows that fully-connected layers are better than SVM in predicting the exact target labels correctly.

Table 3: Classification results of the evaluated models.

Metrics	Before hyper-parameter tuning (%)		After hyper-parameter tuning (%)	
	FC Layers	SVM	FC Layers	SVM
Rank-1	93.99	92.14	94.91	93.81
Rank-5	99.85	99.88	99.85	99.94
Accuracy	93.99	91.90	94.91	93.45
Precision	94.07	91.99	94.93	93.45
Recall	93.99	91.90	94.91	93.45
F1 score	93.88	91.88	95.00	93.44
AUC ROC score	96.79	95.68	97.29	96.51

Matthew's correlation coefficient is measured for both the models when trained with and without hyper-parameter tuning which is shown in Table 4. As discussed in Section 3.4, MCC value close to +1 represent mostly correct predictions. Both models had MCC value close to +1 and the difference between its value for both the models was quite less (0.01∼0.02%). Fully-connected layers outperformed SVM when used both with and without hyper-parameter tuning.

Table 4: Model evaluation results.

Metrics	Before hyper-parameter tuning (%)		After hyper-parameter tuning (%)	
	FC Layers	SVM	FC Layers	SVM
Matthews Correlation Coefficient (MCC)	0.9359	0.9137	0.9457	0.9302

6 Conclusion and Future Work

In this research paper we explored the capabilities of a hybrid model built by combining residual neural networks with either support vector machines (ResNet-152-SVM) or stack of fully-connected layers (ResNet-152-FC) for LULC scene classification. ResNet-152 which is 152 layers deep is used for bottleneck feature extraction from global average pooling layers that returns features of dimension 1x2048. These bottleneck features are fed to either SVM or stack of FC layers for training and testing for classification. Hyper-parameter optimization to further enhance the overall accuracy of the model was applied for both the classifiers. The results gathered showed that ResNet-152-FC with an overall accuracy of 94.91% outperforms the ResNet-152-SVM model with 93.45% overall accuracy for LULC scene classification. The methodology described by Guo et al. (2019) was adapted which was used for LPI radar waveform recognition to implement ResNet-152-SVM with slight modifications in our research. The current ResNet-152-SVM model with an overall accuracy of 93.45% failed to outperform its state-of-the-art performance for waveform recognition with an overall accuracy of 97.8%. This could be due to the fact that both the studies dealt with completely different datasets. Also different layers were chosen for feature extraction from ResNet-152 in both the research.

However, the models implemented in this research showed good classification results, it is suggested to implement land use and land cover classification model for multi-label images also. This will require complex algorithms and more time and effort. It is also suggested to apply hyper-parameter tuning for other important parameters such as number of neurons, activation function for intermediate and last layers and loss function for fully-connected layers classifier. The decision function used in SVM for this research is one-against-one. Further experiments could be performed to explore the effects of using one-against-all decision function for SVM.

7 Acknowledgement

I would like to acknowledge my supervisor, Prof. Noel Cosgrave, for his relentless support and encouragement through every stage of implementation of this project. I would like to take some time to thank, Northwestern Polytechnical University (NWPU) for creating a publicly available remote sensing image dataset for scene classification for research. Lastly I would also like to express my profound gratitude towards my parents and friends for providing an unfailing support and continuous encouragement throughout my years of study.

References

Cao, Y., Xu, R. and Chen, T. (2015). Combining convolutional neural network and support vector machine for sentiment classification, pp. 144–155.

Cheng, G., Han, J. and Lu, X. (2017). Remote sensing image scene classification: Benchmark and state of the art, *Proceedings of the IEEE* **105**(10): 1865–1883.

Copur, M., Ozyildirim, B. M. and Ibrikci, T. (2018). Image classification of aerial images using cnn-svm, *2018 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pp. 1–6.

Elleuch, M., Maalej, R. and Kherallah, M. (2016). A new design based-svm of the cnn classifier architecture with dropout for offline arabic handwritten recognition, *Procedia Computer Science* **80**: 1712 – 1723. International Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA.
URL: *http://www.sciencedirect.com/science/article/pii/S1877050916309991*

Gidudu, A., Hulley, G. and Marwala, T. (2007). Image classification using svms: One-against-one vs one-against-all, *CoRR* **abs/0711.2914**.

Guo, Q., Yu, X. and Ruan, G. (2019). Lpi radar waveform recognition based on deep convolutional neural network transfer learning, *Symmetry* **11**: 540.

Han, S., Kim, M., Lim, W., Park, G., Park, I. and Chang, S. (2018). Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm, *Journal of Investigative Dermatology* **138**.

He, K., Zhang, X., Ren, S. and Sun, J. (2015). Deep residual learning for image recognition, *CoRR* **abs/1512.03385**.
URL: *http://arxiv.org/abs/1512.03385*

Helber, P., Bischke, B., Dengel, A. and Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **12**(7): 2217–2226.

Hu, J., Shen, L. and Sun, G. (2017). Squeeze-and-excitation networks, *CoRR* **abs/1709.01507**.
URL: *http://arxiv.org/abs/1709.01507*

Jannesari, M., Habibzadeh, M., Aboulkheyr, H., Khosravi, P., Elemento, O., Totonchi, M. and Hajirasouliha, I. (2018). Breast cancer histopathological image classification: A deep learning approach, *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2405–2412.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features, *Proceedings of the 10th European Conference on Machine Learning*, ECML'98, Springer-Verlag, Berlin, Heidelberg, pp. 137–142.
URL: *https://doi.org/10.1007/BFb0026683*

Ju, C., Bibaut, A. and Laan, M. (2017). The relative performance of ensemble methods with deep convolutional neural networks for image classification, *Journal of Applied Statistics* **45**.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks, *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1725–1732.

Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks, *Commun. ACM* **60**(6): 84–90.
URL: *http://doi.acm.org/10.1145/3065386*

Liu, Y., Wang, R. and Zeng, Y. (2007). An improvement of one-against-one method for multi-class support vector machine, *2007 International Conference on Machine Learning and Cybernetics*, Vol. 5, pp. 2915–2920.

Nijhawan, R., Sharma, H., Sahni, H. and Batra, A. (2017). A deep learning hybrid cnn framework approach for vegetation cover mapping using deep features, *2017 13th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*, pp. 192–196.

Nogueira, K., Penatti, O. A. B. and dos Santos, J. A. (2016). Towards better exploiting convolutional neural networks for remote sensing scene classification, *CoRR* **abs/1602.01517**.
URL: *http://arxiv.org/abs/1602.01517*

Qi, X., Wang, T. and Liu, J. (2017). Comparison of support vector machine and softmax classifiers in computer vision, *2017 Second International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, pp. 151–155.

Razavian, A. S., Azizpour, H., Sullivan, J. and Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition, *CoRR* **abs/1403.6382**.
URL: *http://arxiv.org/abs/1403.6382*

Son, H., Choi, H., Seong, H. and Kim, C. (2019). Detection of construction workers under varying poses and changing background in image sequences via very deep residual networks, *Automation in Construction* **99**: 27 – 38.
URL: *http://www.sciencedirect.com/science/article/pii/S0926580518305971*

Tang, Y. (2013). Deep learning using support vector machines, *CoRR* **abs/1306.0239**.
URL: *http://arxiv.org/abs/1306.0239*

Too, E. C., Yujian, L., Njuki, S. and Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification, *Computers and Electronics in Agriculture* **161**: 272 – 279. BigData and DSS in Agriculture.
URL: *http://www.sciencedirect.com/science/article/pii/S0168169917313303*

Vakalopoulou, M., Karantzalos, K., Komodakis, N. and Paragios, N. (2015). Building detection in very high resolution multispectral data with deep learning features, *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 1873–1876.

Wolfshaar, J. V. D., Karaaba, M. F. and Wiering, M. A. (2015). Deep convolutional neural networks and support vector machines for gender recognition, *2015 IEEE Symposium Series on Computational Intelligence*, pp. 188–195.

Yang, C., Rottensteiner, F. and Heipke, C. (2018). Classification of land cover and land use based on convolutional neural networks, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* **IV-3**: 251–258.

Yang, J., Kim, N. K. and Kim, H. K. (2018). Se-resnet with gan-based data augmentation applied to acoustic scene classification technical report.

Yang, Y. and Newsam, S. D. (2010). Bag-of-visual-words and spatial extensions for land-use classification, *GIS*.

Zeiler, M. D. and Fergus, R. (2013). Visualizing and understanding convolutional networks, *CoRR* **abs/1311.2901**.
URL: *http://arxiv.org/abs/1311.2901*