V3DI Ensemble Model for High-Accuracy Aerial Scene Classification

K Aditya Shastry* and Reshma Itagi
Department of Information Science, Nitte Meenakshi Institute of Technology, Bangalore, India
**Corresponding author: K Aditya Shastry (adityashastry.k@nmit.ac.in)*

Submitted: 14 Feb 2024 **Accepted:** 29 Nov 2024 **Published:** 27 Mar 2025

Abstract Aerial images are valuable for observing land, allowing detailed examination of Earth's surface features. As remote sensing (RS) imagery becomes more abundant, there is a growing need to fully utilize these images for smarter Earth observation. Understanding large and complex RS images is crucial. Satellite image scenery categorization, which involves labeling images based on their content, has diverse applications. Deep Learning (DL), using neural networks' powerful attribute learning capabilities, has made significant strides in categorizing satellite imagery scenes. However, recent advances in DL for scenery categorization of RS images are lacking. In our study, we employed three transfer learning (TL) models – VGG16, Densenet201 (D-201), and InceptionV3 (IV3) – for classifying aerial images. VGG16 achieved 94% accuracy, while D-201 and IV3 reached 97% accuracy. Combining these models into an ensemble (V3DI ensemble model) improved accuracy to an impressive 99%. This ensemble model combines individual models' classification decisions using majority voting. We demonstrate the efficiency of this approach by showing how ensemble classification accuracy surpasses that of training individual models. Additionally, we preprocess the dataset with a Gabor filter for edge enhancement and denoising to enhance the model's overall performance.

Keywords: aerial image classification, remote sensing, deep learning, transfer learning, ensemble learning.

1. Introduction

A key data source for terrestrial observation, remotely sensed imagery aids in measuring and observing comprehensive features on top of the earth's surface. The number of images is rapidly increasing, leading to a greater demand for research on how to efficiently produce and analyze the majority of these images captured through remote sensing (RS) technology for detailed Earth inspection. Therefore, it is crucial to be able to understand huge and intricate remotely sensed images. Recognizing scenes in aerial photography is a demanding but required task for effective image interpretation, making it a key study topic. Recent high-resolution benchmark data sets have been made widely accessible by researchers from various organizations for the scene classification of RS data. The practical applications of RS scene categorization in urban planning, the discovery of natural hazards, environmental tracking, vegetation mapping, and geospatial item detection have spurred a number of studies over the past few decades. Fig. 1 illustrates how to appropriately classify a scene using RS imageries and established semantic categories.

The purpose of RS image analysis is to visually inspect Earth's surface using data

Fig. 1. Several scene images from the NWPU-RESISC45 dataset. (**a**) Tennis court; (**b**) commercial area; (**c**) river; (**d**) circular farmland; (**e**) palace; (**f**) golf course.

collected from satellites and unmanned aerial vehicles. Categorization of imagery is the backbone of visual recognition systems; after objects have been tagged with meaningful terms, they may be organized according to their conceptual significance. There are several applications of conceptual image organization in machine vision and imagery processing, including video synthesis, scenery evaluation, detecting objects, picture annotation, extraction, and content-driven image interpretation. By categorizing the pixels into different classes, image classification simplifies the data and makes it easier for users to analyze and understand spatial patterns, trends, and relationships. This information can be employed for several uses, like terrestrial usage and ground cover plotting, urban planning, agriculture monitoring, natural resource management, and environmental studies, among others.

In the past few decades, various academics have suggested multiple Deep Learning (DL) algorithms. These algorithms are used for categorizing scenes. They utilize freely available high-resolution standard data sources within this framework. While these models have demonstrated effective precision for smaller datasets, they encounter limitations in extracting features from high-resolution data. Our observation reveals that architectures enhanced with Gabor layers consistently improve robustness over regular models and maintain high generalization performance in tests. To determine the optimal ensemble learning approach, we train multiple Convolutional Neural Network (CNN) models on aerial datasets preprocessed with Gabor filters. Subsequently, we combine these models by averaging their predictions, resulting in improved accuracy. It is through this combination of predictions that we achieve the benefits of ensemble learning. Our research aims to develop an ensemble DL model capable of categorizing and labeling pixels or groups of pixels in satellite or aerial images based on their spectral values. Key objectives include identifying the most suitable pre-trained DL models for image classification based on previous research, constructing predictive models to extract features from high-resolution data and interpret complex patterns and information within the images, and converting raw data into more meaningful information applicable to various applications.

Transfer Learning (TL) involves applying skills learned from previous tasks to a new model for a different task. In deep learning, pre-trained networks are the core form of TL. There are two popular approaches: enhancing pre-trained models and using them as attribute extractors. The idea behind using pre-trained models as attribute extractors is to utilize the parent model with fresh data, replacing only the model's top layer while keeping its weights constant during training. Our research introduces a new framework that combines TL techniques and an ensemble method for Remote Sensing (RS) imagery classification. This framework incorporates the majority voting method and spatial features of landslides to achieve automatic and accurate results. Various organizations have recently made high-resolution benchmark datasets widely accessible for RS data scene classification.

The three main contributions of our research are as follows:

- The proposed ensemble model consists of pre-trained networks like VGG16, InceptionV3, and DenseNet201 that are trained using the datasets RESISC45 and NWPU-RESISC45 to categorize the RS images more accurately. The dataset is preprocessed with a Gabor filter for edge enhancement and denoising. This aids in enhancing the model's general performance.

- Our research offers a thorough analysis of the latest advancements in this area. We examine current benchmark repositories which are easily available to the public and several deep feature learning-based TL algorithms.

- Finally, we evaluate a number of representative methods including the TL-based DL approach and usage of CNN models with different classifiers which were performed with different datasets

The remaining part of this paper is organized as follows. In Section 2 the relared work is discussed. In Section 3 we propose our solution to the aerial scene classification problem. The results of experiments are presented in Section 4. Section 5 concludes the paper.

2. Related work

This section gives a complete evaluation of the works encompassing a broad spectrum of relevant work in RS imagery categorization is presented including network architectures (traditional CNN; Fully Convolutional Networks, FCN; encoder-decoder, recurrent networks; attention models, and generative adversarial models). The characteristics, capabilities, and limitations of current DL models were examined, and potential research directions are discussed. Since RS gives us imagery that enables accurate measurement and study of the Earth's surface features, it is a vital data source for Earth observation. These remote-sensing images offer a wealth of information, allowing us to examine and study the intricate details of our planet's surface. A growing volume of software's use remote sensor pictures. Because of this, it is more important than ever to figure out how to effectively use expanding quantities of RS information for insightful earth observation. Therefore, it is crucial to comprehend large and intricate remote-sensing images. Images captured by satellite have a variety of applications. Advanced learning uses images or data that have been remotely acquired. Consequently, researchers are able to comprehend numerous facets of the relevant domains. The noise and blur present in such photographs is a severe disadvantage. Noise removal is therefore the primary and initial step in the learning of images. Various sounds are added to the images as an outcome of the surroundings, such as particles in the sensor devices, light attenuation, haze, dust, etc. For professionals, pre-processing these photos is a difficult undertaking.

The various noise types [34] in the photos include Saltand-Pepper, Gaussian, Poisson noise, etc. Sudden and quick adjustments in the signal are what generate salt and pepper noise. Errors in the data flow cause this type of noise to arise. It is depicted by pixels of white and black that are sporadically present. The signal is equally spread out over the Gaussian noise. It has a Gaussian distribution and is statistical noise. A function with a probability density function that is identical to the Normal distribution is known as a Gaussian distribution. Each point's vibration operates independently of pixel value intensity. Poisson noise occurs when there aren't enough samples collected by the sensor to generate visible statistical data. Shot noise seems to be present in varying quantities, unlike light and electric current. Only a few of the numerous industries that have effectively exploited RS of images include classification and change detection. Nevertheless, RS image processing involves a few pre-processing processes in addition to categorization and change detection, and it also largely relies on the approach used. Because of this, the RS group is always trying to get better at areas like pre-processing, segmentation, and classification through the use of RS methodologies. The DL (DL) community has long used neural networks, which form the foundation of these techniques. Prior to the development of DL models, ensemble classifiers, including random forests (RF) and support vector machines (SVM), had replaced neural networks as the primary focus in the area of RS for the purposes of picture categorization and additional jobs like change detection. Due to its ease of use (e.g., being largely unaffected by classification parameter sensitivity) and typically high accuracy, SVM has grown in favour (Giorgos Mountrakis, 2011) [26].

While RF acquired popularity for a variety of factors including its capacity to manage high-dimensional data and perform effectively with limited training samples [3]. However, the growth of DL in more recent years has sparked a resurgence in interest in neural networks. DL algorithms have displayed outstanding performance at many image analysing tasks, including object identification, scene classification, and “land use and land cover” (LULC) classification. Thanks to advancements in computer and remote monitoring satellite technology, the images now have better spatial resolution, texture information and appropriate processing techniques. Data from ”High Spatial Resolution RS” (HSRRS) was utilized effectively for information extraction, categorization, and object identification [22, 23, 37]. Numerous HSRRS images have been collected recently, and key work has been done in the domains of pattern recognition, LULC [2, 7]. These methods begin by mining attributes from training information before creating a classification model to test on more data. Most of the recognition techniques are DL-based. DL performs better for target identification, object recognition, and classification. It was effectively used to extract abstract and semantic features [11, 14, 15, 18, 24, 33, 35].

CNNs are a popular DL technique and several CNN-based methods have been designed for Natural Language Processing (NLP), computer vision, processing of medical images, and processing of images from remote sensors [30]. These real-world examples showed that a network’s depth is crucial for the model since it allows it to extract more complicated characteristics when there are more layers. While a deep layered model will improve performance and require a small amount of training, deep CNN (DeCNN) models frequently need a large quantity of labelled data. Finding enough labelled data to train the DeCNN model for the HSRRS scene classification issue is challenging. Additionally, labelling the HSRRS data requires a lot of manpower and materials. When the amount of labelled data is insufficient, the trained DeCNN model will quickly exhibit an over-fitting issue. Many research investigations have proven that TL is effective in classifying and identifying objects from small-sized training records.

2.1. RS imagery scene categorization

- **Multilevel RS imagery scene categorization** An in-depth study has been conducted on the challenge of single-label photo categorization throughout the past few decades. However, because of the usage of the birds-eye imaging method, it is very typical in the actual world for several ground objects to show up in a satellite image. Due to this, single-label RS picture scene categorization hinders the ability to fully comprehend the complex information included in RS imageries. Despite recent research into multilabel satellite image scene classification [4, 9, 12, 13, 19, 31, 32], there are still a number of challenges that require to be resolved, including how to utilize the relationships between various labels, how to acquire more generally applicable discriminative characteristics, and how to construct sizable multilevel datasets for scene categorization.

- **Increasing the size of scene classification datasets** Every type of scene in every open-world scenario would be easily and accurately recognised by the ideal scene classification system. Recent scene classification algorithms are only potentially capable of classifying scenes which prevail in the training datasets because they are trained on a restricted number of datasets. Consequently, a convincing method for classifying scenes ought to appropriately classify a brand-new scene image. There are many fewer scene classes in the published datasets [24–26] than what people can differentiate. A normal deep CNN also has a myriad of criteria and a propensity to overfit the several thousand of training instances. As a consequence, it is very difficult to completely train a deep classification model using the scene classification records that are presently available. Most sophisticated scenery categorization procedures either use pre-trained CNNs as attribute miners or refine already-trained CNNs on the target records. Transfer techniques outperform a fully trained deep CNN model on target records with fewer types and examples, but it's not the best choice because, with enough training samples, the model can derive more specialized characteristics that can adapt to the target domain.
- **RS imageries** The category of RS imageries also includes photographs taken by astronomical satellites. These are employed in a variety of research on the environments of stars, planets, and galaxies. Galaxies' classification is another important topic of research. Astronomers are expected to need vast volumes of data for their studies. The categorization of this material as disks, bars, spirals, etc., will aid in the prediction of the evolution of the cosmos. The bars signify fully developed galaxies and the end of the academic years. In order to research the marine mammals that live in the deep ocean, RS photographs are also used to capture underwater scenes. Additionally, the investigation of oceanic sediment, seas, etc. makes use of these images. They are essential to the investigation of the occurrence of different species in the water. The Catalina survey and the Sloan Digital Sky survey both provide the galaxy view and astronomical data set as open source. These show a telescope-taken screenshot of the sky. Because of how blue and greyscale these telescopic views are, studying the sun and stars is difficult. To learn from these photos, several methods for mistake or noise removal must be incorporated. The categorization of these images is crucial to the domain of physics. Abraham et al. [1] employed the DNN architecture to categorize barred and non-barred galaxies. A 95% accuracy rate in categorizing the images was made possible by the neural network. There are still more advancements to be made in astronomical research, and they will shortly occupy a sizable developmental area. In other words, there hasn't yet been a comprehensive and methodical analysis of how DL is used in the domain of RS.

Although Zhu et al. [38] review was comprehensively compared to others, it focused on a few deserving (and more prevalent) sub-regions related to the domain of RS, such as 3D modeling applications, while disregarding numerous others, such as image classification applications. To fully and objectively comprehend the uses of DL for RS analysis, it seems necessary to conduct a greater amount of structured (i.e., quantitative) study. The number of publications on the subject is clearly growing at the moment. In a range of RS subfields, DL methods are used. For instance, scholars interested in both DL and RS can benefit from understanding the various applications of DL and the problems identified in such studies. Scene classification using remote-sensing data has many applications, which try to assign semantic categories to remote-sensing images based on their contents. Due to DNNs' capacity for feature learning, the categorization of remote-sensing picture scenes has attracted a great deal of attention. The study [6] offers a methodical assessment of DL algorithms for RS picture scenery categorization by encompassing a total of 160 articles, considering the field's quick evolution. The primary methods in categorizing and surveying RS images are auto-encoder-inspired techniques, CNN methods for classifying RS imageries, and "Generative Adversarial Network" (GAN) based methods. The efficacy of several sample techniques/algorithms on three frequently utilized standard datasets is also described in this study [6]. Specifications for RS imagery classification, and new and exciting research avenues are investigated.

2.2. Scene classification for RS images

An image captured by RS includes a range of ground objects. An industrial landscape might, for instance, include buildings, trees, and roads. Scene classification is much more challenging than object-oriented classification due to the diversity and complex spatial distributions of earth's surface objects that prevail in the scene. The classification of satellite picture sceneries has been the subject of extensive research in the past. To group RS imageries with sufficient precision, however, there hasn't yet been a method which can do it. The difficulty with within-class diversity is primarily due to the wide variations in how ground things look across semantic classes. It could be challenging to correctly classify scene images because ground objects frequently vary in design, structure, size, and distribution. Additionally, because of the imaging circumstances, that could be affected by factors like cloud, mist, weather, etc., when RS imageries are taken by aerial platforms, there can be evident variations in hue and rays strength observable inside the identical semantic class. Among the difficulties in classifying RS imageries are the significant within-class diversity and between-class similarity. Within-class diversity may also result from alterations in picture illumination. For instance, in Fig. 2, the physical characteristics of the scene labeled beach exhibit significant variations under various imaging circumstances. When it comes to between class similarity, the main problem is when the same items are present in various scene classes or there exists semantic

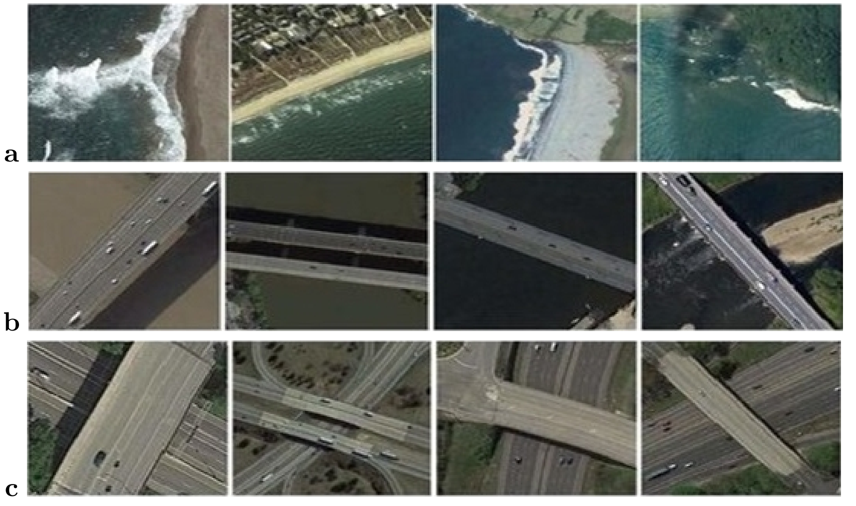


Fig. 2. Images with within-class diversity (**a**) and between-class similarity (**b**) and (**c**) used for scene classification. (**a**) Beach; (**b**) bridge; (**c**) flyover.

overlap between different scene types. As shown in Fig. 2, the bridge scene classes have structures similar to flyover scene classes.

So, both the bridge scenes and flyover scenes could contain identical objects including bridges, and share an abundance of semantic information. Moreover, interclass dissimilarity develops as a consequence of the ambiguous meaning of scene classes. Additionally, some complicated scenes share visual elements with one another. Consequently, it might be very challenging to differentiate between these scene classes. Unsupervised DL models that GANs have lately gained a lot of popularity (Goodfellow et al., 2014 [10]). The GAN has a framework consisting of a generative network and a discriminative network that are in competition with one another. The generative network's generated data and actual data are distinguished by the discriminative network. The generative network gains the ability to map from a hidden space to an interesting data distribution. It is trained with the intention of "fooling" the discriminative network. A convolutional network is typically used by the discriminative network to generate odds. In a zero-sum game, the two networks attempt to optimise a distinct and antagonistic loss function (Oliehoek et al., 2017 [27]). GANs have been effectively utilized in numerous computer vision and image processing applications over the past three years. Based on a hierarchical law method, Classification Tree [16] is a multivariate, incremental, and evolutionary pattern detection system. The root, non-terminal, and terminal of a CT are its preceding components. It determines membership by frequently using binary dividing laws to segment a dataset. These laws are based on "impurity" and are determined using mathematical techniques from the training data. A given node is perfect and there is no impurity present if the pixels that contain it belong to the same group. Unless the logical requirement is satisfied at a specific node, the left branch is selected; in that case, the right branch is noted. Until the terminal node is obtained or till the node is free, the loop continues to run. SVMs, among the most current advancements in AI, are based on ideas from statistical theory. Additionally, SVMs have outperformed the majority of other image classification algorithms in terms of classification accuracy. SVMs are binary classifiers that divide training data into two groups and maximize the difference between them using appropriate hyperplane division on multidimensional function space.

2.3. Recent study on DL techniques used for numerous applications

By comparing the overall accuracy of scene categorization of RS imagery methods using feature extraction done by pre-trained networks and classifiers, collected in Tab. 1, with proposed TL we can observe that the suggested technique of combining multiple pre-trained networks and forming an ensemble model achieves ideal classification performance in most classes.

3. Proposed methodology

It is challenging to classify remote data because of the intricacy of the environment, the variety of remote data, image processing, and classification techniques, to name just a few factors. The dynamic process of RS necessitates numerous picture analyses. Choosing a classification scheme, picking training objects, pre-processing the image, extracting features, putting appropriate classification methods into practice, postclassification storing, and accuracy evaluation are some of the essential actions in the classification of pictures.

TL is a DL technique that involves reusing a previously trained model. TL consists of re-using the knowledge gained from previously seen tasks to be applied to a newly created model in another task. In the context of DL, pre-trained networks are the basic form of TL. The two most popular approaches are using pre-trained models as feature extractors and fine-tuning pre-trained models. The main idea of using pre-trained models as feature extractors is to only replace the top layer of the source model and use it with the new data without updating the model weights during training. We will use this approach of incorporating prior knowledge taken from RS research literature in the creation of a customized high-performing DL model for satellite data tasks. The focus is to utilize a mildly complex and efficiently pre-trained model, learned from a large amount of reference information, such as ImageNet, and then "transfer" the learned information to a relatively simple task (in this study, extract features for scene classification) with a limited amount of information. Three features are very helpful in the transfer:

Tab. 1. Comparative analysis of the proposed method with other state-of-the-art techniques used for image scene classification.

Ref.	Method	Dataset	Accuracy	Description
[29]	DDIPNET DDIPNET+	Aerial Image Dataset-AID	(95.31±0.22)%	Applied the Discriminant Deep Image Prior Network and the Discriminant Deep Image Prior Network+, which combine Deep Image Prior and Triplet Networks learning strategies.
[36]	The suggested technique creates an SVM-based RS picture classifier	Aerial images from Pingshuo mining area, China	94.72%	After using the seven-layer CNN model, whose activation function makes use of the TReLU function, the three high-level image features are sequentially fused and fed into the SVM classifier.
[21]	TLDeCNN	High Spatial Resolution Remote Sensing Scenes (HSRRS)	96.1%-VGG19, 97.1%-ResNet50, 99.4%-InceptionV3	The three TL-DeCNN models TLVGG19, TLResNet50, and TLInceptionV3 are put forth for use in classifying HSRRS scenes in built-up urban regions.
[28]	CNN + PCA + SVM	UC Merced and WHU-RS	(98.26±0.40)%	With the extraction of features, a unique representation of the features was produced by combining the average pooling layer's features and the convolutional layer's PCA transformed features with SVM for classification.
[20]	DNN, i.e., CNN, CapsNet, SMDTR-CNN, and SMDTR-CapsNet	HSRRS	95.0%	Convolutional neural networks (CNN), capsule networks (CapsNet), the same model based on CNN with a different training rounding (SMDTR-CNN), and the same model based on CapsNet with a different training rounding are the four deep neural networks (DNNs) used. (SMDTR-CapsNet).
[25]	Applied DCNN Model	High resolution satellite image	92.0%	Multichannel water body detection network.
[17]	FeatSpace EnsNets and Avg EnsNets for Change Detection	ESA's Sentinel-1 and DInSAR processed georeferenced images	84.862%	Applied TL by fine-tuning neural networks (TLFT) is referred to as "Feature space Ensembles of TLFE neural networks" (FeatSpaceEnsNets) and "Average Ensembles of TLFE neural networks" (AvgEnsNets).
[8]	TL-DenseUNet	Remote Sensing Images dataset from NSFC, China	92.4%	There are two subnetworks in Applied TL DenseUNet. A transferring DenseNet pre-trained on three-band ImageNet images is one of them that the encoder subnetwork employs.

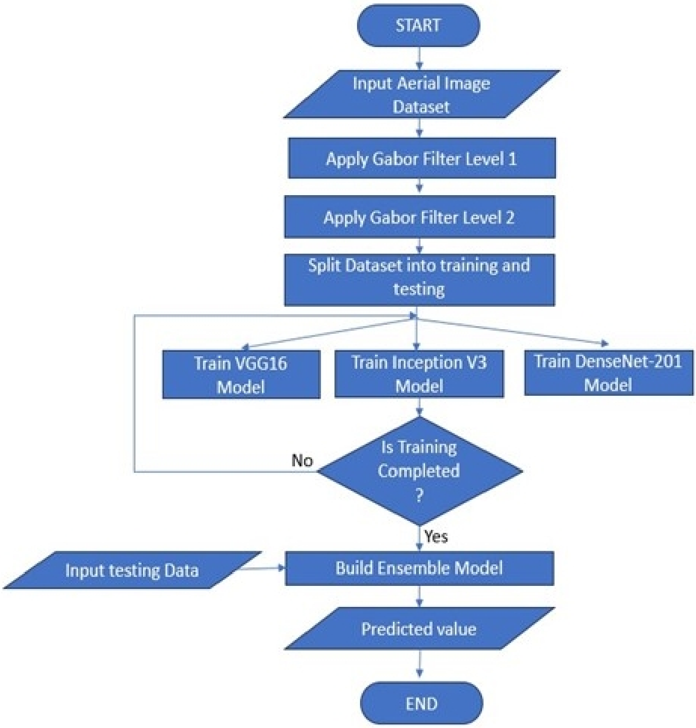


Fig. 3. Flowchart for the proposed classification procedure for RS images

- The advancement of the pre-trained model eases the process of getting rid of hyper-parameter tuning.
- The very first layers of a pre-trained model could be assumed to feature extractors, assisting in the separation of minimal features such as edges, colors, masses, shades, and surfaces.
- Because we accept that the subsequent layers complete the complicated identifying tasks, the objective framework may only need to retrain the next few layers of the pre-trained model.

The dynamic process of RS necessitates numerous picture analyses. Fig. 3 shows the flowchart of the classification procedure used for RS images.

It is challenging to classify remote data because of the intricacy of the environment, the variety of remote data, image processing, and classification techniques, to name just a few factors. Choosing a classification scheme, picking training objects, pre-processing the image, extracting features, putting appropriate classification methods into practice,

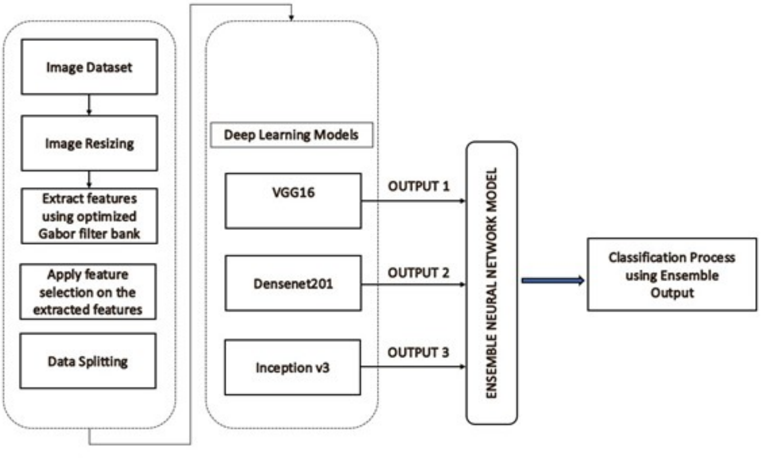


Fig. 4. Architecture of the proposed Ensemble Model for the automatic classification of RS images

post-classification storing, and accuracy evaluation are some of the essential actions in the classification of pictures. In this work, we stack multiple pre-trained CNN models on the ImageNet repository as an Ensemble Model and use them to train the aerial datasets. We train the ensemble model to get a high validation accuracy of around 98%. The output is obtained by combining the predictions of multiple trained models and averaging them. The proposed model is illustrated in Fig. 4.

The models are then combined by averaging, there taking a weighted average increases accuracy. The advantage of the ensemble learning method, which is improved predictive performance, is attained by combining the predictions. Ensemble techniques of stacking the model decisions rely on the assumption that different models will not make the same mistakes. In other words, given a set of diverse models, minimum of one will make the right classification. Fig. 5 demonstrates the data flow chart of the devised work.

TL has developed as a powerful method in the domain of DL, allowing us to leverage pre-trained models that were on large-scale datasets. By using TL, we could profit from the learned features of these models and adapt them to our specific task of aerial image classification. In this study, we employ three well-known TL models: VGG16, Densenet201, and InceptionV3. By combining the outputs of these models we aim to leverage their diverse strengths and improve the overall accuracy of aerial image classification. The findings and perceptions derived from this research will contribute to the advancement of automated analysis and understanding of aerial imagery, benefiting a wide range of applications and industries.

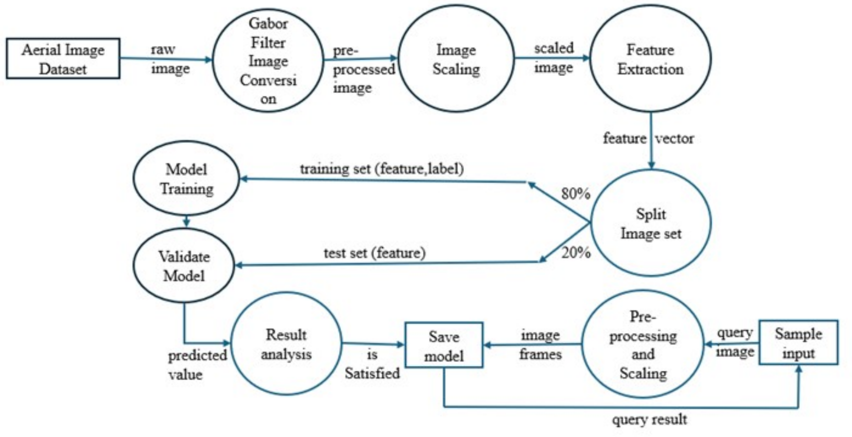
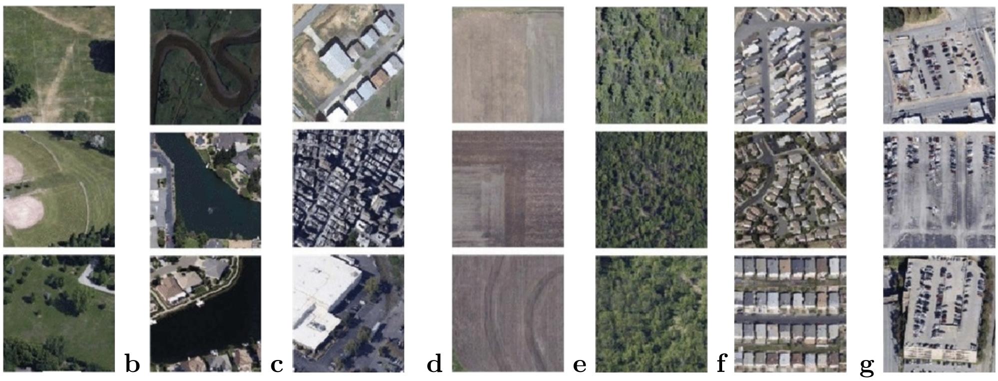


Fig. 5. Data flow plot of the devised work

3.1. Image acquisition and preprocessing

In this module, we will get the data from the online source. Further, we will resize the image for future use. Image resizing, also known as imagery scaling, is a geometrical operation that applies an image approximation technique to alter the scale of an image. By increasing or decreasing the pixel density of a target image, the aforementioned resizing operation modifies the final dimension of image information. Computers are able to perform computations on 0s and 1s and are unable to interpret images in the way that we do. We have to somehow convert the images to 0s and 1s for the computer to understand. The image will be converted to grayscale (range of gray shades from white to black) the computer will assign each pixel a value based on how dark it is. All the 0s and 1s are put into an array and the computer does computations on that array. We then feed the resulting array for the next step. It is common practice to first divide the dataset into two parts, the “Train” and the “Test”. Then, with the Test set put aside, the Train set is selected at arbitrarily from the Train dataset, and the remaining (100-X)% is used as the Validation set, with X resolved at a certain percentage (e.g., 80%). The algorithm is then developed and verified continually using these two sets. So we will follow the same method to prepare data for the training and testing phase. We are building our model by using the Ensemble network. Now that we’re done pre-processing, we can start implementing our individual deep-learning model. Max-pooling: A technique used to reduce the dimensions of an image by taking the maximum pixel value of a grid. This also helps reduce overfitting and makes the model more generic. After that, we add 2 fully connected (FC) layers. Since the input of FC layers should



a b c d e f g

Fig. 6. Predefined classes from RSSCN7 dataset. (**a**) grass; (**b**) river; (**c**) industrial; (**d**) field; (**e**) forest; (**f**) residential; (**g**) parking.

be two-dimensional, and the convolution layer output is four-dimensional, we need a flattening layer between them. At the very end of the FC layers is a softmax layer. We are using 2 remote-sensing image datasets in this study.

- NWPU-RESISC45 Dataset: It is a 2017 dataset presented by Cheng, Han, and Lu [5]. It contains 31 500 256×256 RGB images split into 45 classes of 700 images each. It represents a comprehensive dataset regarding land use. NWPU-RESISC45 has a more diverse assortment of classes as well as more samples per class. The authors developed it in order to have both urban and rural classes as well as scenes classifiable by large features and small features alike. For all models, the input images are resized to 64×64×3, resizing from their original size of 256×256x3 while preserving the 3 RGB channels.
- RSSCN7 Dataset: It contains satellite imageries learned from Google Earth, which is originally collected for RS imagery categorization [39]. We conduct image synthesis on RSSCN7 to make it capable of the image inpainting task. It has seven classes: "grassland, farmland, industrial and commercial regions, river and lake, forest field, residential region, and parking lot". Each class has 400 images, so the dataset comprises of 2 800 images. Samples of the dataset are shown in Fig. 6.

When processing images, the linear Gabor filter is just one of several methods used for things like feature extraction, examination of texture, boundary identification, etc. The band pass filtering devices, of which gabor filters are a subset, allow through only a narrow range of frequencies (or "band") while blocking out all others. A Gabor filter can be viewed as a sinusoidal signal of particular frequency and orientation, modulated by a Gaussian wave. Gabor descriptor for an image is computed by passing the image through a filter bank of Gabor filters. Gabor filter represents a linear band-pass filter whose impulse response is defined as a Gaussian function modulated with a complex sinusoid. An artificial and actual part, signifying opposite directions, make up the filter's structure. The two components may be formed into a complex number.

$$g(x,y;\lambda,\theta,\phi,\gamma)=\exp\left(-\frac{x'^2+\gamma^2y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda}+\phi\right), \tag{1}$$

where λ – wavelength of the sinusoidal component, θ – direction of the normal to the parallel bands of Gabor function, ϕ – phase balance of the sinusoidal procedure, σ – standard deviation of the Gaussian envelope, and γ – the Gabor function's ellipticity specified by the temporal dimension ratio.

The artificial and actual components of the "Gabor filter kernel" are applied to the image and the response is returned as a pair of arrays. An example of a linear filter is the "Gabor filter", which uses a sinusoidal plane waveform as modulation and a Gaussian base. Similar to how human vision represents frequencies and direction, the Gabor filter does the same. Gabor filters are appropriate for edge detection and texture classification. Attributes are mined from gray-scale character images by Gabor filters which are particularly devised from numerical data of character forms. An adaptive sigmoid module is employed to the outputs of Gabor filters to achieve better performance on low-quality images.

Gabor filters are used in texture analysis, edge detection, and attribute mining. When a Gabor filter is applied to an image, it delivers an optimal outcome at boundaries and at points where texture changes. Algorithm 1 illustrates the procedure used to generate Gabor layers for the input dataset.

In algorithm 1, two output directories (`gdir_1` and `gdir_2`) are defined to store the processed images. The function `create_dir()` is called to ensure that the required directories exist. This function is responsible for creating directories if they do not already exist. The algorithm iterates through each `label` in the input directory (`in_dir`). Each `label` corresponds to a category or class of images. For each `label`, the algorithm processes each image in the corresponding directory. If the image file does not conclude with `'.db'` (indicating it is not a database file), the following steps are performed:

- The path to the current image (`in_path`) and the output paths for the processed images (`out_path_1` and `out_path_2`) are defined.
- The image is read from `in_path`.
- Two Gabor kernels (`gabor_1` and `gabor_2`) are generated with specified parameters.
- The image is filtered using `gabor_1` to obtain `filtered_img_1`, and then `filtered_img_1` is further filtered using `gabor_2` to obtain `filtered_img_2`.
- The resulting filtered images (`filtered_img_1` and `filtered_img_2`) are saved to the corresponding output paths (`out_path_1` and `out_path_2`).

Once all images for a particular label have been processed, the algorithm ends.

Algorithm 1 Resource Allocation Algorithm

```
1: Initialize directories:
     Set gdir_1 as '/content/drive/MyDrive/Colab Notebooks/processed/gabor_1'
     Set gdir_2 as '/content/drive/MyDrive/Colab Notebooks/processed/gabor_2'
2: Create directories:
     Call create_dir(in_dir, out_dir, gdir_1, gdir_2)
3: Process images:
     for each label in the list of directories in in_dir do
       for each image_name in the list of files in the dir corresponding to label do
         if image_name does not end with '.db' then
           Set in_path as the path to the current image
           Set out_path_1 as the output path for the first Gabor filtered image
           Set out_path_2 as the output path for the second Gabor filtered image
           Read the image img from in_path
           Generate Gabor kernel gabor_1 with parameters:
             Size: (18, 18)
             Sigma: 1.5
             Theta: π/4
             Lambda: 5.0
             Gamma: 1.5
           Filter img using gabor_1 to obtain filtered_img_1
           Write filtered_img_1 to out_path_1
           Generate Gabor kernel gabor_2 with the same parameters as gabor_1
           Filter filtered_img_1 using gabor_2 to obtain filtered_img_2
           Write filtered_img_2 to out_path_2
         end if
       end for
   end for
```

Algorithm 1 essentially preprocesses a collection of images using Gabor filters. It enhances features in the images by applying Gabor filtering with two different kernels (`gabor_1` and `gabor_2`). The administered imageries are subsequently kept in separate directories (`gdir_1` and `gdir_2`). The final goal of this preprocessing step is to prepare the images for further processing or analysis, such as imagery categorization using TL models like VGG16, Densenet201, InceptionV3 and the ensemble model.

After preprocessing of the input dataset with Gabor filter the features in the image are enhanced and help in training the model to distinguish between pre-defined categories. Fig. 7 shows sample imageries which are acquired after preprocessing with the Gabor filter bank.

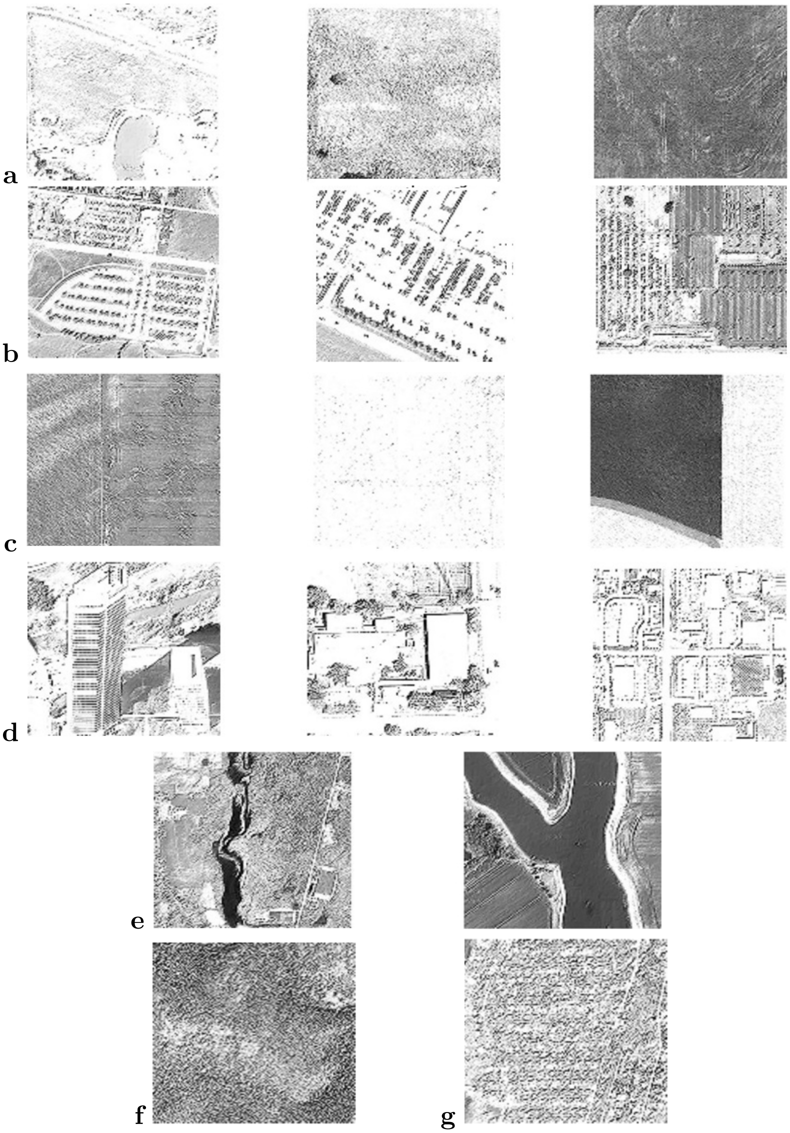


Fig. 7. Sample scene images categorized under different classes processed by the Gabor filter from the NWPU-RESISC45 dataset. (**a**) Grass; (**b**) parking; (**c**) field; (**d**) industry; (**e**) river lake; (**f**) forest; (**g**) resident.

3.2. Ensemble model

In machine learning, the ensemble approach refers to combining multiple models to improve the general performance of the prediction task. The fundamental notion in ensemble learning is that by combining multiple models, we can reduce the risk of individual models making errors and enhance the model's generalization performance. Ensemble methods can be applied to a wide range of machine learning algorithms, including decision trees, neural networks, and support vector machines. There are two key kinds of ensemble learning techniques: Bagging: Several models are trained independently on different subsets of the training data, and their predictions are aggregated by taking the average (for regression problems) or voting (for classification problems) of the outputs. Boosting: Here, the models are trained sequentially, and each novel model is trained to rectify the errors of the previous models. The notion is to create a strong model by combining many weak models. This system uses the Bagging ensemble approach. Ensemble learning integrates the forecasts from numerous neural network models to decrease the changes in forecasts and decrease generalization errors. The method starts by fine-tuning the 3 CNN models using the training data. By comparing models like VGG16, InceptionV3, and DenseNet201 and combining its output to get higher accuracy among the model predictions. To amalgamate the projected outcomes, the most straightforward approach is to compute the mean of the forecasts generated by the base classifiers.

4. Experimental results

This Section provides details concerning the results obtained during the classification of data with NWPU-RESISC45 and RSSCN7 datasets. It also summarizes the diverse measures utilized for performance evaluation. When utilizing DL methods for image classification, choosing the appropriate evaluation metrics is essential for determining which model to use, how to adjust the hyperparameters, whether regularization approaches are necessary, and other issues. Tab. 2 in page 43 summarizes the evaluation metrics for four models (Ensemble, InceptionV3, DenseNet201,and VGG16) across seven target classes (field, forest, grass, industry, parking, resident, and river lake).

The models generally exhibit high performance, with high precision, recall, and F1-scores for most classes. The Ensemble model achieves near-perfect scores for all classes, while the other models also show strong performance, although with some variations across metrics and classes. Overall, the models demonstrate accurate classification of the target classes, highlighting their effectiveness. A confusion matrix is a way to summarize the performance of a classification model by comparing its predictions with the actual values of a dataset. It is a square matrix where the rows correspond to the true or actual classes, and the columns correspond to the predicted classes. Accuracy is a measure that generally describes how the model performs across all classes. (number of all correct predictions divided by the total number of elements in the dataset).

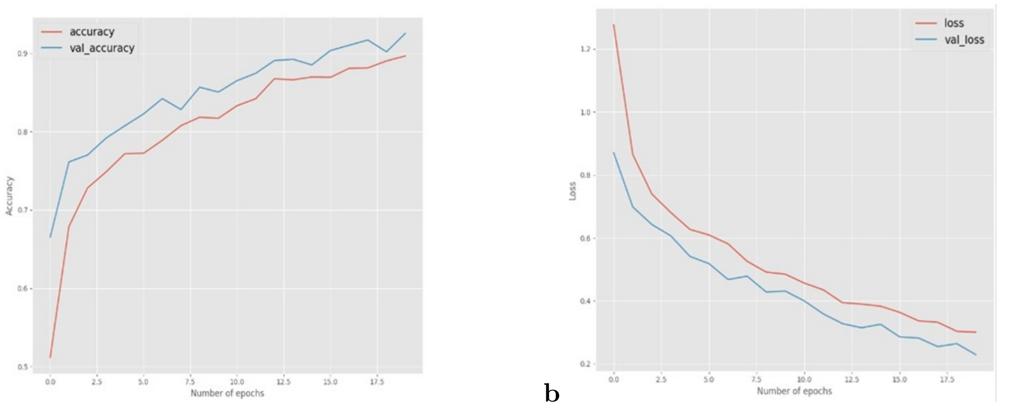
Tab. 2. Classification report of pre-trained models with ensemble model.

VGG16 Model Results				
Class	Precision	Recall	F1-Score	Support
Field	0.92	0.97	0.94	69
Forest	0.99	0.99	0.99	78
Grass	0.94	0.92	0.93	50
Industry	0.87	0.97	0.92	63
Parking	0.97	0.85	0.90	66
Resident	0.95	0.97	0.96	62
River lake	0.98	0.93	0.96	60

Densenet201 Model Results				
Class	Precision	Recall	F1-Score	Support
Field	1	0.99	0.99	69
Forest	0.95	1	0.97	78
Grass	0.94	1	0.97	50
Industry	0.91	0.98	0.95	63
Parking	1	0.88	0.94	66
Resident	0.97	0.94	0.95	62
River lake	1	0.98	0.99	60

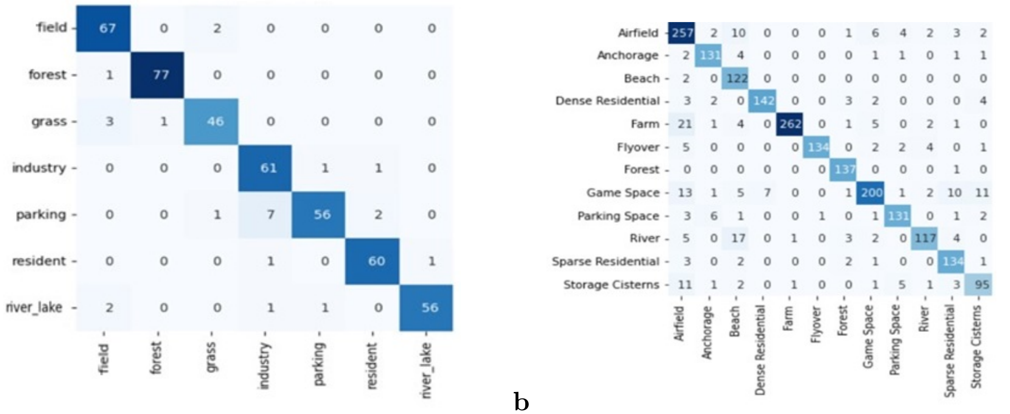
InceptionV3 Model Results				
Class	Precision	Recall	F1-Score	Support
Field	0.92	1	0.96	69
Forest	1	0.97	0.99	78
Grass	1	0.96	0.98	50
Industry	0.98	0.92	0.95	63
Parking	0.93	0.97	0.95	66
Resident	0.95	0.98	0.98	62
River lake	0.9	0.93	0.97	60

Proposed Ensemble Model Results				
Class	Precision	Recall	F1-Score	Support
Field	1	1	1	69
Forest	1	1	1	78
Grass	0.98	0.98	0.98	50
Industry	0.98	0.98	0.98	63
Parking	1	1	1	66
Resident	1	1	1	62
River lake	1	0.98	0.99	60



a b

Fig. 8. Training model accuracy and loss plot graphs on RSSCN7 using using VGG16. (**a**) Accuracy plot graph; (**b**) Loss plot gragh.



a b

Fig. 9. VGG16 confusion matrix for the classification result for: (**a**) RSSCN7 dataset; (**b**) NWPU-RESISC45 dataset.

4.1. Experimental findings on the VGG16 Model

Fig. 8 demonstrates the training model accuracy and loss plot graphs on RSSCN7 using VGG16.

Fig. 9 shows the chart for the confusion matrix VGG16 model.

From the confusion matrix it can be concluded that this model is performing excellent. 448 photos from RSSCN7 dataset are provided in total and this model achieved 94% accuracy.

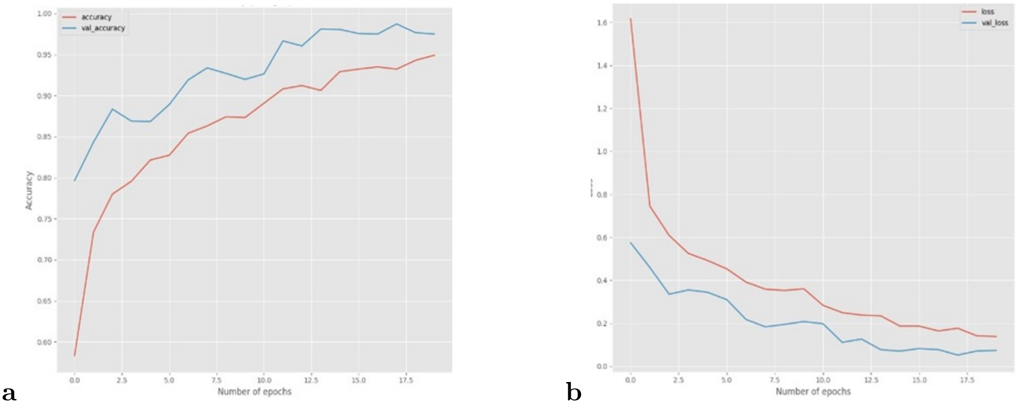


Fig. 10. Training model accuracy and loss plot graphs on RSSCN7 using DenseNet201. (**a**) Accuracy plot graph; (**b**) Loss plot gragh.

4.2. Experimental findings on the DenseNet201 model

Fig. 10 illustrates the training model accuracy and loss plot graphs on RSSCN7 using DenseNet201.

Fig. 11 displays the confusion chart for the Densenet201 model, illustrating its performance. The confusion matrix indicates that the model performs exceptionally well. A total of 448 photos from RSSCN7 dataset were evaluated, and the model achieved an impressive accuracy rate of 97%.

4.3. Experimental findings on the InceptionV3 model

Fig. 12 shows the training model accuracy and loss plot graphs on RSSCN7 using InceptionV3.

Fig. 13 illustrates the confusion matrix chart for the InceptionV3 model. Centered on the examination of the confusion matrix, it is evident that the model's performance is excellent. The dataset consisted of a total of 448 photos from RSSCN7 dataset, and the model achieved an impressive accuracy of 97%.

4.4. Experimental findings on the ensemble approach

An ensemble model of InceptionV3, Densenet201, and VGG16 was evaluated using a confusion matrix, represented by Figure 14. With 444 out of 448 images correctly predicted for the RSSCN7 dataset, the resulting accuracy is 99.11%, indicating high accuracy.

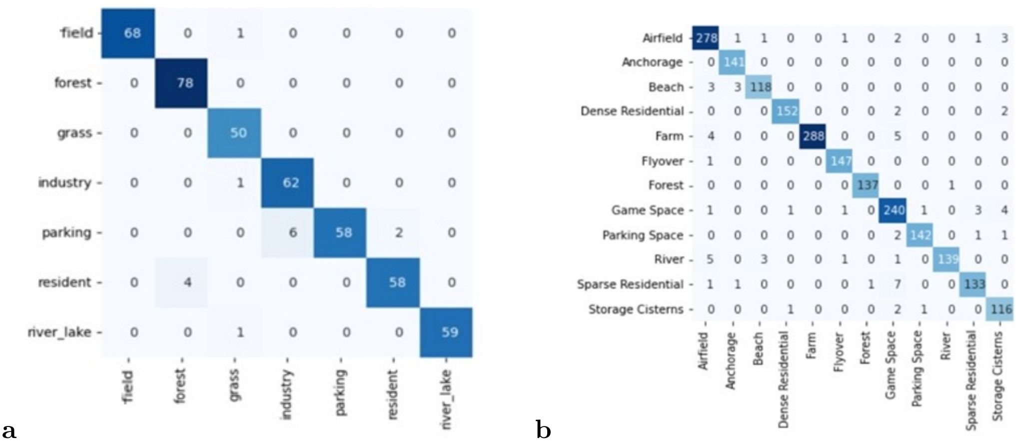


Fig. 11. DenseNet confusion matrix for the classification result for (**a**) RSSCN7 dataset; (**b**) NWPU-RESISC45 dataset.

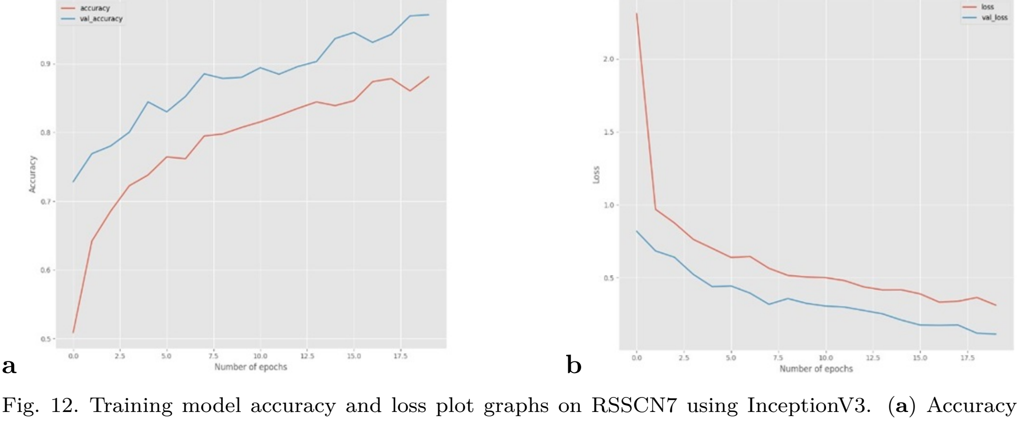


Fig. 12. Training model accuracy and loss plot graphs on RSSCN7 using InceptionV3. (**a**) Accuracy plot graph; (**b**) Loss plot graph.

The ensemble model performed exceptionally well in predicting the labels for these images. Fig. 14 shows the training model accuracy and loss plot graphs on RSSCN7 using InceptionV3.

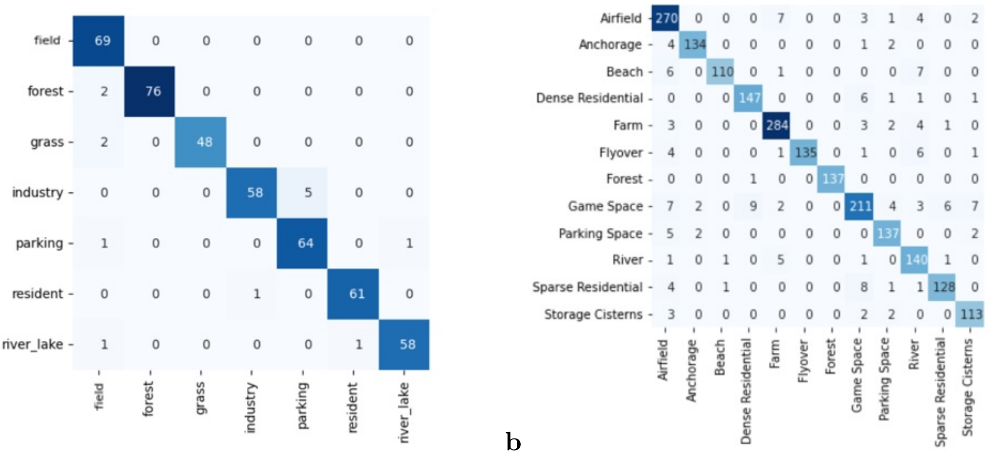


Fig. 13. InceptionV3 confusion matrix for the classification result for: (**a**) RSSCN7 dataset; (**b**) NWPU-RESISC45 dataset.

Fig. 14. Ensemble Confusion matrix for the classification result for: (**a**) RSSCN7 dataset; (**b**) NWPU-RESISC45 dataset.

4.5. Ensemble technique results comparision and analysis

In this study on aerial image classification, TL and ensemble models were employed to achieve high accuracy. Three individual models, VGG16, Densenet201, and InceptionV3, were evaluated, and an ensemble approach was utilized. Results showed that Densenet201 achieved an accuracy of 97.1%, while InceptionV3 performed even better

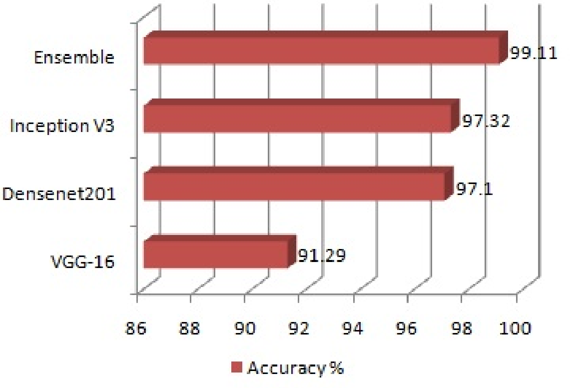


Fig. 15. Accuracy comparison graph for RSSCN7 dataset.

with 97.32%. VGG16, although slightly worse, still attained a respectable accuracy of 91.29%. Fig. 15 shows the graph depicting the compared models.

The ensemble approach, combining all three models, yielded a remarkable accuracy of 99.11%. These findings highlight the effectiveness of TL in improving classification accuracy, with InceptionV3 being the most successful individual model. The ensemble approach demonstrated its strength by surpassing the accuracy of all individual models. Overall, this research contributes to the advancement of aerial imagery study, delivering significant insights for various applications such as urban planning, environmental monitoring, and agriculture.

5. Conclusion and future scope

The various RS classification techniques are fully described in this paper, along with the study fields where they are used. The different algorithms employed in the classification and clustering of the RS images and how this data is compiled for the learning process are also covered in detail. It also goes into great depth on how to extract features from remote-sensing images using a CNN Model. This study has developed a DL-based classification method for RSSCN7 and NWPU-RESISC45 datasets with multiscene classes. The analysis of these datasets using Gabor filters and TL has achieved high accuracy in classifying diverse forms of aerial view scenes. Specifically, VGG16 attained an accuracy of 94%, DenseNet201 achieved 97%, and InceptionV3 also had 97%. These high levels of accuracy suggest that the Gabor filters and DL models are efficient in mining important attributes from aerial view images and accurately classifying them. An ensemble model built with these 3 models was also evaluated with sample test data. The accuracy, kappa coefficient, F1 score, and confusion matrix have all been used to verify the performance of the suggested approaches. The results revealed that the ensemble model obtained the best overall accuracy (99.11%) and kappa coefficient (0.99), while also improving the precision of river samples by 4%, respectively. The model's accuracy is valuable for a range of applications, such as RS, weather forecasting, and environmental monitoring. By identifying various aerial view scenes with high accuracy, the model offers effective intuitions into ecological surroundings, that would aid to inform decision-making processes. For instance, the model can help predict weather patterns, monitor air quality, and assess changes in environmental conditions.

Further research and testing will be necessary to evaluate the model's generalizability and robustness across different datasets and image categories. It's important to note that the model's performance is limited to the specific dataset and image categories utilized in the analysis. It might not generalize appropriately to new datasets or different types of aerial view scenes. Therefore, further research is needed to test the model's robustness and generalizability across different datasets and applications. The sample size and scope of the study could be increased, as well as the variation in aerial view scenes captured. Additionally, the study focused on the accuracy of the model, but it is also important to consider other performance metrics to gain a more complete insight of the model's effectiveness. Future extensions could include fine-tuning individual pre-trained networks. We also need to look into new methods and systems which may be utilised to deploy the combination of RS information from social media, and spatial technology in order to advance the state-of-the-art of RS image scene categorization.

References

[1] S. Abraham, A. Aniyan, A. Kembhavi, and N. Philip. Detection of bars in galaxies using a deep convolutional neural network. *Monthly Notices of the Royal Astronomical Society* 477(1):894–903. 2017. doi:10.1093/mnras/sty627.

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

[3] M. Belgiu and L. Drăguţ. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing* 114:24–31. 04 2016. doi:10.1016/j.isprsjprs.2016.01.011.

[4] G. Cheng, D. Gao, Y. Liu, and J. Han. Multi-scale and discriminative part detectors based features for multi-label image classification. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 649–655. 2018. doi:10.24963/ijcai.2018/90.

[5] G. Cheng, J. Han, and X. Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE* 105:1865–1883. 04 2017. doi:10.1109/JPROC.2017.2675998.

[6] G. Cheng, X. Xie, J. Han, K. Li, and G.-S. Xia. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13:3735–3756. 2020. doi:10.1109/JSTARS.2020.3005403.

[7] G. Cheng, C. Yang, X. Yao, K. Li, and J. Han. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs. *IEEE Transactions on Geoscience and Remote Sensing* 56(5):2811–2821. 2018. doi:10.1109/TGRS.2017.2783902.

[8] B. Cui, X. Chen, and Y. Lu. Semantic segmentation of remote sensing images using transfer learning and deep convolutional neural network with dense connection. *IEEE Access* 8:116744–116755. 2020. doi:10.1109/ACCESS.2020.3003914.

[9] B. Demir and L. Bruzzone. A novel active learning method in relevance feedback for content-based remote sensing image retrieval. *IEEE Transactions on Geoscience and Remote Sensing* 53(5):2323–2334. 2015. doi:10.1109/TGRS.2014.2358804.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, et al. Generative adversarial networks. *Communications of the ACM* 63(11):139–144. 2020. doi:10.1145/3422622.

[11] G. Gui, H. Huang, Y. Song, and H. Sari. Deep learning for an effective non-orthogonal multiple access scheme. *IEEE Transactions on Vehicular Technology* 67(9):8440–8450. 2018. doi:10.1109/TVT.2018.2848294.

[12] Y. Hua, L. Mou, and X. Zhu. Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification. *ISPRS Journal of Photogrammetry and Remote Sensing* 149:188–199. 2019. doi:10.1016/j.isprsjprs.2019.01.015.

[13] Y. Hua, L. Mou, and X. X. Zhu. Relation network for multilabel aerial image classification. *IEEE Transactions on Geoscience and Remote Sensing* 58(7):4558–4572. 2020. doi:10.1109/TGRS.2019.2963364.

[14] H. Huang, Y. Peng, J. Yang, W. Xia, and G. Gui. Fast beamforming design via deep learning. *IEEE Transactions on Vehicular Technology* 69(1):1065–1069. 2019. doi:10.1109/TVT.2019.2949122.

[15] H. Huang, J. Yang, Y. Song, H. Huang, and G. Gui. Deep learning for super-resolution channel estimation and doa estimation based massive mimo system. *IEEE Transactions on Vehicular Technology* 67(9):8549–8560. 2018. doi:10.1109/TVT.2018.2851783.

[16] H. Jiang, D. Zhao, Y. Cai, and S. An. A method for application of classification tree models to map aquatic vegetation using remotely sensed images from different sensors and dates. *Sensors* 12(9):12437–12454. 2012. doi:10.3390/s120912437.

[17] Z. Karim and T. van Zyl. Deep learning and transfer learning applied to sentinel-1 dinsar and sentinel-2 optical satellite imagery for change detection. In: *2020 International SAUPEC/RobMech/PRASA Conference*, pp. 1–7. 2020. doi:10.1109/SAUPEC/RobMech/PRASA48453.2020.9041139.

[18] N. Kato, Z. Fadlullah, B. Mao, F. Tang, O. Akashi, et al. The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective. *IEEE Wireless Communications* 24(3):146–153. 2017. doi:10.1109/MWC.2016.1600317WC.

[19] N. Khan, U. Chaudhuri, B. Banerjee, and S. Chaudhuri. Graph convolutional network for multi-label vhr remote sensing scene recognition. *Neurocomputing* 357:36–46. 2019. doi:10.1016/j.neucom.2019.05.024.

[20] W. Li, H. Liu, Y. Wang, L. Zhuangzhuang, Y. Jia, et al. Deep learning-based classification methods for remote sensing images in urban built-up areas. *IEEE Access* 7:36274–36284. 2019. doi:10.1109/ACCESS.2019.2903127.

[21] W. Li, Z. Wang, Y. Wang, J. Wu, J. Wang, et al. Classification of high-spatial-resolution remote sensing scenes method using transfer learning and deep convolutional neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13:1986–1995. 2020. doi:10.1109/JSTARS.2020.2988477.

[22] H. Liu, X. Huang, F. Han, J. Cui, B. Spencer, et al. Hybrid polarimetric GPR calibration and elongated object orientation estimation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13:1986–1995. 2019. doi:10.1109/JSTARS.2019.2912339.

[23] H. Liu, H. Xia, M. Zhuang, Z. Long, C. Liu, et al. Reverse time migration of acoustic waves for imaging based defects detection for concrete and CFST structures. *Mechanical Systems and Signal Processing* 117:210–220. 2019. doi:10.1016/j.ymssp.2018.07.011.

[24] B. Mao, F. Tang, Z. Fadlullah, N. Kato, O. Akashi, et al. A novel non-supervised deep-learning-based network traffic control method for software defined wireless networks. *IEEE Wireless Communications* 25(4):74–81. 2018. doi:10.1109/MWC.2018.1700417.

[25] M. Mesvari and R. Shah-Hosseini. Segmentation of electrical substations using deep convolutional neural network. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* X-4/W1-2022:495–500. 2023. doi:10.5194/isprs-annals-X-4-W1-2022-495-2023.

[26] G. Mountrakis, J. Im, and C. Ogole. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing* 66(3):247–259. 2011. doi:10.1016/j.isprsjprs.2010.11.001.

[27] F. A. Oliehoek, R. Savani, J. Gallego-Posada, E. Van der Pol, E. D. De Jong, et al. GANGs: Generative adversarial network games. arXiv, arXiv:1712.00679. 2017. doi:10.48550/arXiv.1712.00679.

[28] B. Petrovska, E. Zdravevski, P. Lameski, R. Corizzo, I. Štajduhar, et al. Deep learning for feature extraction in remote sensing: A case-study of aerial scene classification. *Sensors* 20(14):3906. 2020. doi:10.3390/s20143906.

[29] D. F. Santos, R. G. Pires, L. A. Passos, and J. P. Papa. DDIPNet and DDIPNet+: Discriminant deep image prior networks for remote sensing image classification. In: *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pp. 2843–2846. 2021. doi:10.1109/IGARSS47720.2021.9554277.

[30] Z. Shao, J. Cai, P. Fu, L. Hu, and T. Liu. Deep learning-based fusion of Landsat-8 and Sentinel-2 images for a harmonized surface reflectance product. *Remote Sensing of Environment* 235:111425. 2019. doi:10.1016/j.rse.2019.111425.

[31] Z. Shao, W. Zhou, X. Deng, M. Zhang, and Q. Cheng. Multilabel remote sensing image retrieval based on fully convolutional network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13:318–328. 2020. doi:10.1109/JSTARS.2019.2961634.

[32] R. Stivaktakis, G. Tsagkatakis, and P. Tsakalides. Deep learning for multilabel land cover scene categorization using data augmentation. *IEEE Geoscience and Remote Sensing Letters* 16(7):1031–1035. 2019. doi:10.1109/LGRS.2019.2893306.

[33] F. Tang, Z. Fadlullah, B. Mao, and N. Kato. An intelligent traffic load prediction based adaptive channel assignment algorithm in SDN-IoT: A deep learning approach. *IEEE Internet of Things Journal* 5(6):5141–5154. 2018. doi:10.1109/JIOT.2018.2838574.

[34] R. Verma and J. Ali. A comparative study of various types of image noise and efficient noise removal techniques. *International Journal of Advanced Research in Computer Science and Software Engineering* 3:617–622. 2013.

[35] Y. Wang, M. Liu, J. Yang, and G. Gui. Data-driven deep learning for automatic modulation recognition in cognitive radios. *IEEE Transactions on Vehicular Technology* 68(4):4074–4077. 2019. doi:10.1109/TVT.2019.2900460.

[36] B. Xia, F. Kong, J. Zhou, X. Wu, and Q. Xie. Land resource use classification using deep learning in ecological remote sensing images. *Computational Intelligence and Neuroscience* 2022:7179477. 2022. doi:10.1155/2022/7179477.

[37] Y. Zhong, X. Han, and L. Zhang. Multi-class geospatial object detection based on a position-sensitive balancing framework for high spatial resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* 138:281–294. 2018. doi:10.1016/j.isprsjprs.2018.02.014.

[38] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, et al. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine* 5(4):8–36. 2017. doi:10.1109/MGRS.2017.2762307.

[39] Q. Zou, L. Ni, T. Zhang, and Q. Wang. Deep learning based feature selection for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters* 12(11):2321–2325. 2015. doi:10.1109/LGRS.2015.2475299.

K Aditya Shastry completed his Ph.D. in Computer Science and Engineering in 2019 from VTU, India. He completed his B.E. in 2000 and M.Tech. in 2007. He has more than three years of industry experience and 20+ years of teaching experience. He has authored numerous papers in reputed journals and conferences. He has more than 10 book chapters to his credit that are published by reputed publishers. He has given many invited talks as a resource person. He is also a reviewer of reputed journals. To his credit, he has two funded projects from VGST and ISRO. His areas of interest include machine learning, deep learning, and data mining. Currently he is working as Professor in the Department of Information Science and Engineering at Nitte Meenakshi Institute of Technology, Bengaluru, India.

Reshma Itagi received her master's degree from Nitte Meenakshi Institute of Technology, India. She is now a software engineer at Harman, specializing in full-stack development. Her main research interests include machine learning, deep learning, and data analytics.