TerraClassify: Deep Learning-based Land Cover Classification and Semantic Segmentation for Sustainable Development and Automation

Dharu Piraba Muguntharaman, Raunak Bhupal, Shamyukta Rajagopal
New York University
Github Repository: https://github.com/raunakbhupal/land-classification

Abstract

Land cover classification plays a vital role in land use planning and decision-making by accurately categorizing satellite image pixels into specific land cover categories. The goal of this research is to create a deep aggregation network that will serve as a baseline model for land cover categorization and to investigate various methods and models to improve its accuracy. The primary dataset, the DeepGlobe challenge, focuses on classifying land types for industrial and agricultural growth. For sustainable development and effective resource management, understanding and charting land usage are essential. The suggested study uses deep learning approaches to categorize various types of land cover samples. We use CNN models, mainly Unet, with several variants to achieve accurate classification using the DeepGlobe Land Cover Classification Dataset, which comprises structured and information-rich satellite pictures. By providing a comparative examination of various models and methodologies, this study enhances the subject of land cover classification for better land use planning and environmental management.

Introduction

This project focuses on the crucial task of automatically categorizing and segmenting land cover for applications in sustainable development, agriculture, forestry, and urban planning [1]. The availability of publicly accessible datasets has provided opportunities for researchers to address environmental challenges and enhance their understanding of the surrounding environment.Satellite data is a valuable source of information, yet its utilization for machine learning purposes, particularly in the domain of image analysis, remains relatively unexplored. For this project, we have obtained the dataset from the DeepGlobe Challenge [1]. This dataset comprises satellite images along with their corresponding masks, which facilitate the classification of the images into one of the seven land cover categories.

The primary objective of this project is to perform land cover classification using various models and techniques. By building a robust learning model with this dataset, we aim to accurately classify satellite images. The accurate categorization achieved through semantic segmentation is crucial for sustainable development, urban planning, and agricultural applications. Through in-depth analysis and a focused approach to this field, we can significantly improve our understanding of the environment and optimize its utilization as a valuable and limited resource.

Related work

The DeepGlobe challenge, which intends to enhance the field of satellite image analysis through the development and evaluation of deep learning algorithms, is described in the document [1]. Land cover categorization, road extraction, and building detection are the three tasks that make up the challenge. Participants must categorize pixels or regions in satellite photos into predetermined land cover types like vegetation, water, and urban areas. Convolutional neural networks (CNNs), recurrent neural networks (RNNs), and ensemble methods are among the deep learning approaches used. Insights into cutting-edge techniques, evaluation procedures, and areas for development are provided, with an emphasis on the necessity of reliable algorithms for accurate and insightful interpretation of the Earth's surface from satellite photos.

The accuracy and dependability of land usage mapping are the primary objectives of the study [2] . They put forth a novel strategy that enhances the mapping result by utilizing remote sensing data, geographic information systems (GIS), and machine learning approaches.The authors then go into their suggested methodology, which combines advanced machine learning techniques, GIS datasets, and remote sensing data from satellites or other aerial platforms. supervised classification algorithms such random forests, support vector machines, or deep learning models are some of the machine learning methods used in the study. The labeled samples used to train these algorithms are taken from either existing land use maps or ground truth data.

A novel deep learning architecture created exclusively for land cover classification tasks is introduced in the paper [3] . By efficiently extracting spatial and contextual data from remote sensing pictures, the proposed Deep Aggregation Net (DAN) seeks to enhance the precision and efficacy of land cover classification.The architecture of the Deep Aggregation Net, which consists of numerous layers and is intended to capture both local and global contextual data, is shown in this work. Convolutional and pooling layers are included in the network for feature extraction, while aggregation modules are used to efficiently combine data from various spatial scales. The paper also includes discussions on model optimization, hyperparameter tuning, and computational efficiency considerations. The importance of the aggregation modules and the advantages of combining both local and global contextual information are highlighted by the authors as they further investigate the impact of various DAN components using ablation study .

The study [4] investigates the use of deep transfer learning techniques for classifying land use and land cover. The authors perform a comparison analysis to assess how well various transfer learning techniques improve categorization accuracy.The authors give a general review of transfer learning and the various techniques it uses, such as feature extraction and fine-tuning. They describe how starting points for land use and land cover classification tasks can be pre-trained models like VGG, ResNet, or Inception.The authors examine how the performance of classification is affected by elements like dataset properties, model designs, and the volume of transferred information.The authors offer perceptions into how learnt representations are interpreted and transferable across different geographic regions and imaging sensors.

The authors in the paper [5] use the U-Net architecture, a popular deep learning model known for its efficiency in semantic segmentation tasks . The Lovász-Softmax loss function, which is intended to manage class imbalance and optimize for the intersection over union (IoU) metric, is also introduced. The authors describe the encoder-decoder network with skip links that makes up the U-Net architecture. This architecture makes it possible to capture both high-level and low-level information, making it easier to localize land cover classes precisely. Additionally, they discuss the implementation specifics, such as the activation function selection, batch normalization, and data augmentation methods. The authors go further over the benefits of the proposed method, such as its ability to handle class imbalance and successfully divide up small and fragmented land cover regions.

Dataset Description

The dataset used in this project is sourced from the Land Cover Classification Track in the DeepGlobe Challenge on Kaggle. The training data comprises 803 satellite imagery samples in RGB format, with each image having dimensions of 2448x2448 pixels. These images were captured by DigitalGlobe's satellite, offering a pixel resolution of 50cm.

Each satellite image in the dataset is associated with a corresponding mask image that provides land cover annotations. The mask image is also represented in RGB format and contains labels for seven different classes, identified through color-coding (R, G, B) as follows:

Figure 1: Satellite images

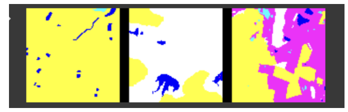

Figure 2: Mask images

- Urban land: 0,255,255 - Represents man-made, built-up areas with human artifacts, excluding roads.
- Agriculture land: 255,255,0 - Encompasses farms, planned plantations, cropland, orchards, vineyards, nurseries, and ornamental horticultural areas, as well as confined feeding operations.
- Rangeland: 255,0,255 - Represents non-forest, non-farm green land areas covered with grass.
- Forest land: 0,255,0 - Represents areas covered by forests.
- Water: 0,0,255 - Represents rivers, oceans, lakes, wetlands, and ponds.
- Barren land: 255,255,255 - Indicates barren areas such as mountains, rocky terrains, deserts, beaches, and areas devoid of vegetation.
- Unknown: 0,0,0 - Represents clouds and other unidentified elements.

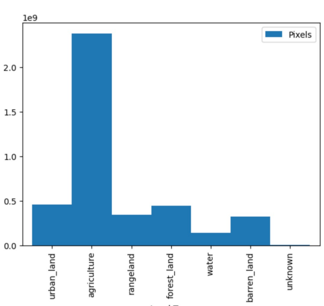

Figure 3: Distribution of classes

It is clear from Figure 3 that the dataset's distribution of images among its various classifications is unbalanced. There's a class imbalance issue in the dataset. The satellite images and their corresponding mask images follow a naming convention, where the satellite image file is named as "image_id_sat.jpg" and the mask image file is named as "image_id_mask.png". The "image_id" portion of the file name is a randomized integer assigned to each image. It is important to note that the values in the mask image may not be limited to pure 0 and 255, as there could be variations within the color range for each class label.

Base Model

For implementation, the first step is to load the dataset and process the satellite and mask images separately. To handle the computational load of working with large images, we employ a technique called patchifying the image, which involves breaking the images into smaller patches [2]. This allows for independent processing of each patch, distributing the workload and reducing memory requirements.

Due to compute constraints, we select a subset of the dataset, specifically 300 images out of the total 683. The metadata provided in the data description indicates that the mask values are of RGB type, so we convert each mask image to RGB format before creating patches. Additionally, we generate labels for the classes based on the metadata description.

Next, we split the training data into train and validation sets, preparing the dataset for feeding into the Unet model. The Unet model, which follows a specific architecture [7], is utilized. It is an encoder-decoder architecture comprising two main sections: the contracting section, which down samples the input image using valid padding convolution layers followed by a ReLU activation function and max pooling layers, and the expanding section, which up samples the image using valid convolutions with ReLU activation and 2x up sampling convolutions. To address

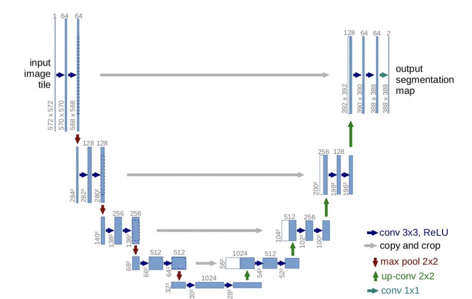

Figure 4: Unet model architecture

the dataset's imbalance issue, we employ Dice loss, a commonly used loss function for image segmentation tasks. Additionally, we calculate focal loss, which focuses the learning process on hard misclassified examples. The total loss is computed using a combination of Dice loss and focal loss.

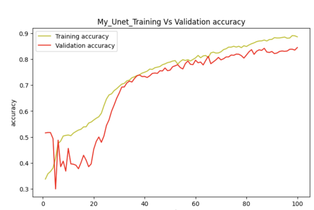

Figure 5: Accuracy of Multi Unet model

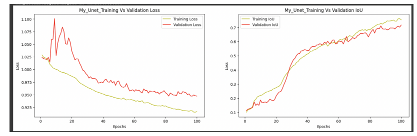

Figure 6: Loss and IoU of Multi Unet model

Once all the parameters are defined and the model architecture is set, we proceed with training the model for 100 epochs. During training, the model achieves an accuracy of 91.66% on the training dataset and 84.51% on the validation dataset.

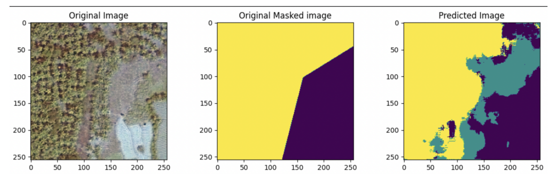

Figure 7: Predictions by Multi Unet model

Ablation Study

In this section, we examine the effects of using different models as a backbone for the basic model. We evaluate the performance of the updated models in comparison to the base model by changing the backbone architecture. The general setup and settings of the models stay the same as the base model unless otherwise specified. This enables us to assess the impact of various backbone models on results and derive conclusions.

ResNet

Here, we use the ResNet34 model as the backbone architecture for the U-Net model with regards to image segmentation.The U Net model uses ResNet34 as its backbone and benefits from its deep architecture and skip connections, which allow it to collect complex features and preserve spatial data during segmentation. The addition of ResNet34 as the model's backbone increases the amount of trainable parameters compared to the base U-Net model because ResNet34 has more layers and connections, which gives it more flexibility when extracting features from the input data. We could see that this model performs better with an increased accuracy of 90%.

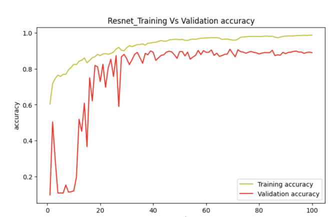

Figure 8: Accuracy of ResNet34 model

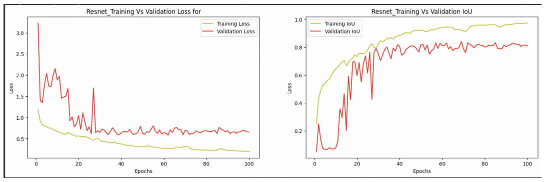

Figure 9: Loss and IoU of ResNet34 model

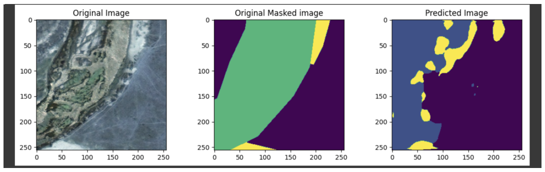

Figure 10: Predictions by ResNet34 model

Inceptionv3

We replace the ResNet34 model with Inceptionv3 as the backbone architecture here. The model can initialize with learnt features and reach faster convergence by utilizing Inceptionv3 and the pre-trained weights from the "imagenet" dataset. As expected there's an increase in the number of trainable parameters because the more complicated architecture of the Inceptionv3-based UNet model allows the model to learn more precise representations and capture finer features.Overall, the UNet model's capacity, feature extraction, and accuracy are all improved by employing Inceptionv3 as the model's backbone by utilizing pre-trained weights. We could see that by training the model for 100 epochs with a batch size of 16 with Adam Optimiser yielded a accuracy of around 93%.

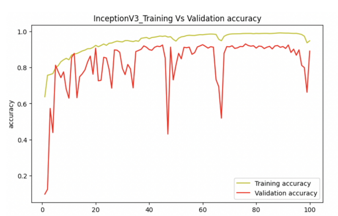

Figure 11: Accuracy of Inceptionv3 model

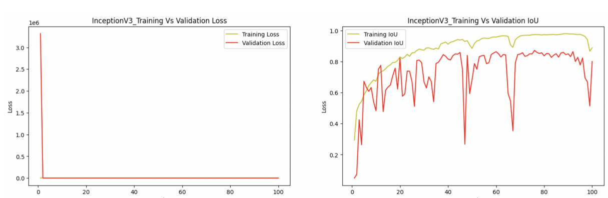

Figure 12: Loss and IoU of Inceptionv3 model

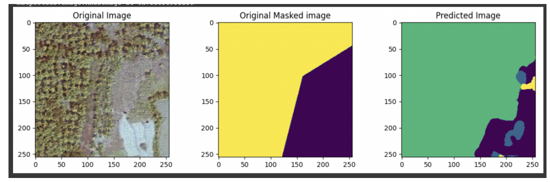

Figure 13: Predictions by Inceptionv3 model

VGG19

Similarly we tried the VGG19 model as backbone architecture to the U-Net model and analyzed the results. It could be seen that the number of trainable parameters increased compared to the base model. The accuracy also increased compared to the base model but it was less compared to the previous 2 models. This difference may be due to the different architecture of the various models.

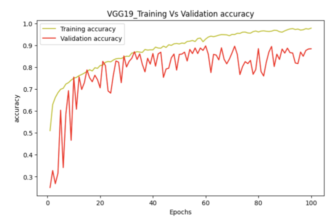

Figure 14: Accuracy of VGG19 model

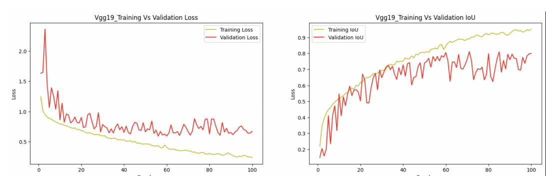

Figure 15: Loss and IoU of VGG19 model

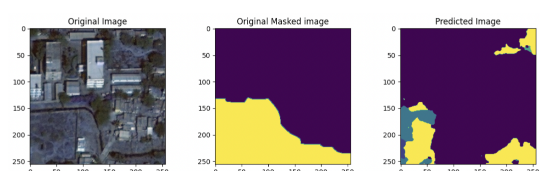

Figure 16: Predictions by VGG19 model

Results

In this study, we experimented with the Unet model using various backbone designs for classifying land cover. By training models for 100 epochs with a batch size of 16, enabling enough training iterations for convergence, the goal was to improve model performance. The base model initially had an accuracy of around 85%. We investigated other backbone architectures, such as ResNet34, Inceptionv3, and VGG19, to enhance this. The VGG19 model among them

Table 1: Model Accuracies

Model	Accuracy (%)
Unet model	85.41
ResNet34 Backbone	90.92
Inceptionv3 Backbone	93.06
VGG19 Backbone	89.76

demonstrated the highest accuracy of about 89%, closely followed by the ResNet34 model at 90%. Surprisingly, the Inceptionv3 model, used as the basic model's architecture, produced the highest accuracy, at around 93%. These findings show how various backbone architectures can improve the accuracy of classifying land cover.

Optimal Model

We could see that the Unet model with inceptionv3 as backbone performs better compared to the other models.The Inceptionv3 architecture is renowned for its effective parameter management and capacity for the detection of intricate patterns in images. The model gains from the pre-trained weights gained by training on the ImageNet dataset by using Inceptionv3 as the backbone.These weights give the network a decent initialization and aid in improving its performance on a variety of computer vision tasks. The Inceptionv3 UNet model's greater accuracy increases the possibility that it was able to more accurately extract the fine details and semantic information from the input photos, leading to more accurate predictions for the classification of land cover.

Table 2: Model Details

Model	Inception v3
Loss Function	Combination of BCE and Jaccard Loss
Learning Rate	0.001
Batch Size	16
Optimizer	Adam

Future Work

The primary objective of this project was to achieve accurate land cover classification by exploring different models and optimizing their performance using various optimizers. The Unet model, combined with the Adam optimizer and Inceptionv3 as the backbone, yielded a notable accuracy of 93.06%. However, due to limitations in computational resources (specifically, training on Google Colab), we encountered restrictions in training the entire dataset, as it exceeded the available RAM capacity. Consequently, we had to experiment with different dataset sizes to ensure feasibility within the given constraints. As a result, we were unable to conduct all the initially planned experiments.

Moving forward, with the availability of adequate computational resources, our future work would involve training the model on the complete dataset. Additionally, we intend to explore alternative optimizers, learning rates, and different backbone architectures. The literature review conducted during the project provided valuable insights into other architectures implemented in various research papers, and we aim to incorporate and evaluate these architectures in our model.

Conclusion

By investigating several backbone architectures for the Unet model, our study's overall goal was to improve the accuracy of land cover classification. To find out which of three popular models namely ResNet34, Inceptionv3, and VGG19 would be most efficient at enhancing classification performance, we conducted a comparative analysis of them. With each backbone architecture displaying distinct advantages, our results showed significant improvements over the base model. With an accuracy of over 93% among them, the Inceptionv3 model stood out to be the best under given parameters. Our initial goal was to further explore the effects of various optimizers and hyperparameter tuning on model performance, however due to computational limitations, we were unable to do so. However, our study has successfully shown that utilizing a variety of backbone architectures can lead to considerable increases in the accuracy of land cover classification. To further improve the performance of these models, future research can concentrate on tweaking hyperparameters and exploring other optimization strategies.

References

[1] Demir, I. et al. (2018) 'DeepGlobe 2018: A Challenge to parse the Earth through satellite images', 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) [Preprint]. doi:10.1109/cvprw.2018.00031.

[2] Improving the mapping outcome of land usage (no date). Available at: http://cs230.stanford.edu/projects_fall_2022/reports/89.pdf.

[3] Kuo, T.-S. et al. (2018) 'Deep aggregation net for land cover classification', 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) [Preprint]. doi:10.1109/cvprw.2018.00046.

[4] Naushad, R., Kaur, T. and Ghaderpour, E. (2021) 'Deep Transfer Learning for Land Use and land cover classification: A comparative study', Sensors, 21(23), p. 8083. doi:10.3390/s21238083.

[5] A. Rakhlin, A. Davydow and S. Nikolenko, "Land Cover Classification from Satellite Imagery with U-Net and Lov asz-Softmax Loss," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 2018, pp. 257-2574, doi: 10.1109/CVPRW.2018.00048.

[6] Baratam, S. (2021) Land cover classification with U-Net, Medium. Available at: https://baratam-tarunkumar.medium.com/land-cover-classification-with-u-

net-aa618ea64a1b (Accessed: 14 May 2023).

[7] Zhang, J. (2019) UNet line by line explanation, Medium. Available at: https://towardsdatascience.com/unet-line-by-line-explanation-9b191c76baf5 (Accessed: 14 May 2023).

[8] Land cover classification using satellite imagery and Deep Learning (no date) ArcGIS API for Python. Available at: https://developers.arcgis.com/python/samples/land-cover-classification-using-unet/ (Accessed: 14 May 2023).

[9] Garg, L. et al. (2019) 'Land use land cover classification from satellite imagery using munet: A modified unet architecture', Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications [Preprint]. doi:10.5220/0007370603590365.

[10] Agarwal, R., Goel, S. and Nijhawan, R. (2022) 'Using deep learning approach for land-use and land-cover classification based on satellite images', 2022 2nd Asian Conference on Innovation in Technology (ASIANCON) [Preprint]. doi:10.1109/asiancon55314.2022.9909395.