Deep Learning - Land Cover Classification

Fabris Daniele
2060942
Deep Learning 2024/25
Università di Padova - DEI

Abstract

The project aims to develop a deep learning system that analyzes images captured by Sentinel-2 satellites, extracting information from multiple bands, to maximize classification performance.

During development, several architectures were tried, ranging from custom CNNs, vision transformers, ensemble networks, etc., up to a hybrid model combining a 3D CNN network with a transformer encoder.

Introduction

Land cover classification is a critical task in environmental applications, such as agriculture, disaster recovery, climate change, urban development, or environmental monitoring that require an accurate analysis of multispectral data to distinguish between different land classes. Sentinel-2 data offer high spatial and spectral resolution, but the complexity of band relationships requires advanced models. This work proposes a hybrid architecture that leverages the ability of 3D CNNs to analyze spectral and spatial features, combined with the flexibility of Transformers to model contextual relationships.

The EuroSAT large-scale patch-based land use and land cover classification dataset was used for this study, comprising multispectral images of 13 bands representing different land use classes. The dataset is based on Sentinel-2 satellite images covering 13 spectral bands and consists of 10 classes with a total of 27,000 labeled and geo-referenced images. Each image includes 13 channels, corresponding to the following spectral bands: Aerosols, Blue, Green, Red, Red Edge 1, Red Edge 2, Red Edge 3, Near Infrared (NIR), Red Edge 4, Water Vapor, Cirrus, Short Wave Infrared (SWIR) 1, Short Wave Infrared (SWIR) 2.

By utilizing this extensive set of channels, I was able to incorporate richer spectral information beyond traditional RGB values. This approach significantly improved the model's predictive accuracy.

Method Overview

Data Preprocessing

Data preprocessing is a crucial step to ensure that the model receives consistent and representative inputs of the problem at hand. The main preprocessing steps applied in the project are described below:

1. Image Loading - The EuroSAT dataset is composed of multispectral images in ".tiff" format, each with 13 spectral bands representing different features of the Earth's surface.
2. Band Statistics Calculation - For each spectral band, the mean and standard deviation over the entire dataset were calculated. These values were used to normalize each band.
3. Normalization - Each pixel of each band was transformed according to the formula: where is the mean and is the standard deviation of the band. This step ensures that the bands have a distribution with mean 0 and standard deviation 1, making them comparable and suitable for the model.
4. EuroSATTransform Pipeline - The transformation pipeline defined in the project includes the following operations:
   - Random Flip: introduces variations by randomly flipping images horizontally and vertically.
   - Random Rotation: introduces variations by randomly rotating images by 90, 180, 270 degrees.
   - Random Crop and Resize: a random crop is performed to increase spatial variability in the training data and then images are resized to a standard size to ensure consistent inputs to the model.
   - Consistent Resizing: The images were resized to a uniform size of 64x64 pixels to ensure compatibility with the model.
   - Random Gaussian Noise: random Gaussian noise is added to improve the robustness of the model to unexpected variations in the data.
5. Data to Tensor Conversion - Images are converted to PyTorch tensors to be compatible with the framework. This step includes transposing the dimensions from (H, W, C) to (C, H, W), where is the number of channels.

These steps ensure that the data is pre-processed to take full advantage of the multispectral information in the dataset, increasing the model's chances of success in the land cover classification task.

Proposed Method

The project employs a hybrid approach for the classification of multispectral images from the EuroSAT dataset. The main objective is to leverage the rich information provided by the 13 spectral bands and combine it with the representational power of a deep learning model.

The method consists of two primary components:

1. Spatial Feature Extraction with a 3D CNN - This component is designed to analyze the spatial and spectral patterns present in the multispectral data using 3D convolutions.
2. Feature Integration with a Transformer Encoder - After feature extraction by the CNN, a custom Transformer Encoder is used to capture complex and long-range relationships among the extracted features.

This design allows the model to harness both the spatial locality of CNNs and the global dependency modeling capability of Transformers.

Model Architecture

The model architecture, named HybridModel, consists of the following components:

1. 3D Convolutional Neural Network (CNN) - Takes as input the multispectral images with 13 bands.
   - Conv3D Layers
     - First layer: 13 input channels, 64 output channels, kernel size (3, 3, 3), stride 1, and padding 1.
     - Second layer: 64 input channels, 128 output channels, similar kernel and stride settings.
     - Third layer: 128 input channels, 256 output channels, same settings as before.
   - Activation and Normalization - Each convolution is followed by BatchNorm3D and ReLU.
   - Pooling
     - MaxPooling3D with a kernel size of (2, 2, 2) to reduce dimensions while preserving spatial context.
     - Final AdaptiveAvgPooling3D reduces the output to a fixed spatial dimension (1, 16, 16).
   - Output - Flattened feature vector with dimensions 256 * 16 * 16.
2. Fully Connected Layer for CNN Output
   - Reduces the flattened vector's dimensionality to the embedding size (512).
   - Output Shape - (batch_size, 512).
3. Transformer Encoder - Takes as input a single sequence with 512 features.
   - Multi-head Attention - 8 attention heads enable the model to attend to multiple feature representations simultaneously.
   - Feed-forward Network - Fully connected layers with a hidden dimension of 1024.
   - Layer Normalization and Dropout - Improve stability and prevent overfitting.
   - Repeated for 6 layers to enhance representation.
4. Final Fully Connected Classifier - Takes as input the flattened Transformer output.
   - Output - Probabilities for each of the 10 classes.

This HybridModel combines localized feature learning (CNN), capturing local details and spatial structure in multispectral images which are crucial for distinguish between land cover types, with global feature aggregation (Transformer), modeling long-range relationships among features and enhancing the model's generalization capability. Thanks to the modularity of the pipeline, the architecture is adaptable to datasets with varying number of bands or resolutions. Furthermore, the combination of CNN and Transformer makes the model particularly suited for multispectral data, which often includes highly correlated bands.

Training and Testing

Hardware

The computer on which the tests are performed is equipped with: Processor AMD Ryzen 7 6800h with Radeon graphics, Installed RAM 2x16GB DIMM DDR5 4800MHz, System type 64-bit operating system, NVIDIA GeForce RTX 3070 8GB GDDR6, SSD 1TB M2.2 2280 PCIe Gen4 TLC.

Training, Validation and Testing

The training process was carried out on two distinct splits of the dataset: an 80-20 split and a 50-50 split. The 80-20 split represents 80% of the data used for training and validation combined, and 20% reserved for testing. Similarly, the 50-50 split divides the data equally between the training-validation set and the testing set.

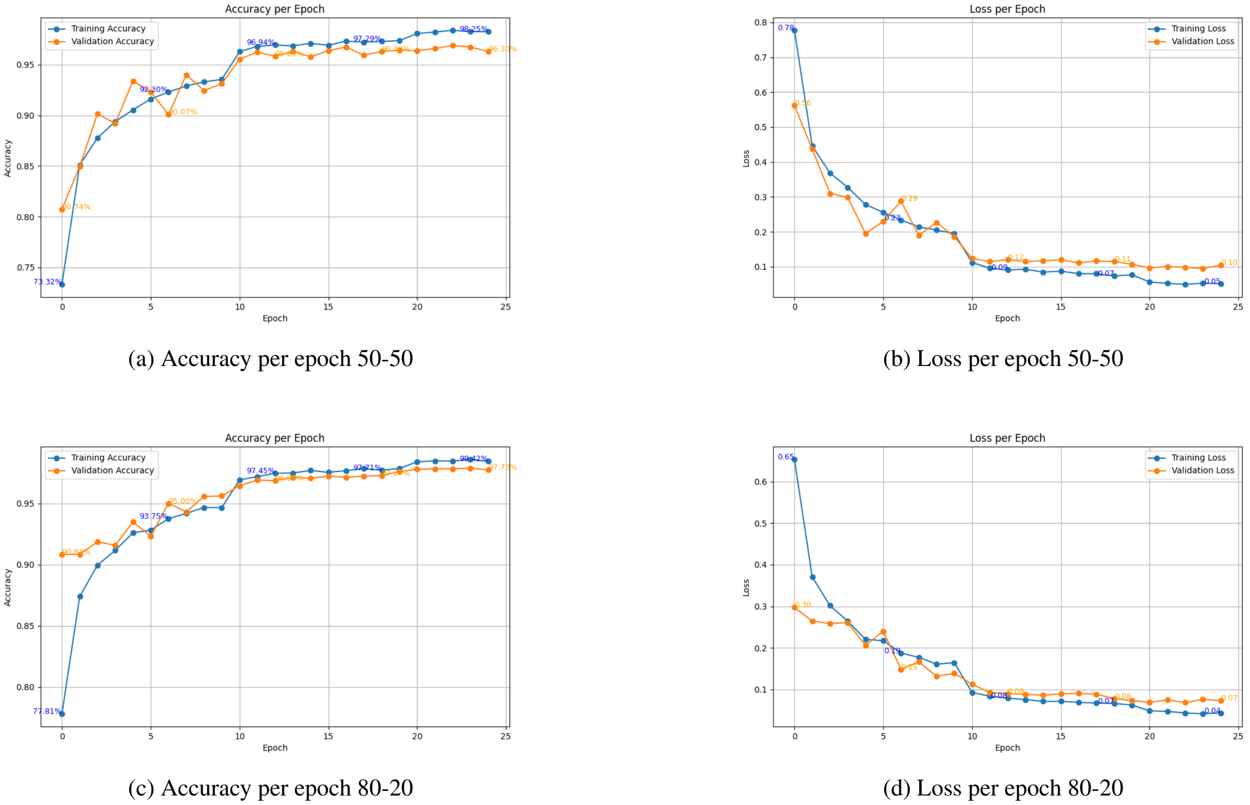
For both splits, 10% of the training set was further set aside as a validation set to monitor model performance during training and prevent overfitting. This validation set played a crucial role in hyperparameter tuning and early stopping.

The training pipeline leveraged the HybridModel architecture, which combines convolutional layers for feature extraction and transformer layers for capturing long-range dependencies. The model was optimized using the Adam optimizer with a learning rate scheduler to adjust the learning rate dynamically based on validation loss. The criterion used was Cross-Entropy Loss, ensuring compatibility with multiclass classification tasks.

Validation was performed on the 10% subset of the training set, as mentioned above. Metrics such as accuracy, precision, recall, and F1-score were evaluated at each epoch to assess the model's performance and guide hyperparameter adjustments. Regular validation allowed for:

- Monitoring of the model's ability to generalize.
- Detection of overfitting by comparing training and validation metrics.
- Fine-tuning of the learning rate and other hyperparameters.

Testing was conducted on the reserved portions of the dataset (20% for the 80-20 split and 50% for the 50-50 split). This phase provided an unbiased evaluation of the model's



(a) Accuracy per epoch 50-50

(b) Loss per epoch 50-50

(c) Accuracy per epoch 80-20

(d) Loss per epoch 80-20

Figure 1: Accuracy and Loss per epoch for the two training splits 50-50 and 80-20.

performance on unseen data. Key metrics evaluated during testing included:

- Overall accuracy - The percentage of correctly classified samples.
- Class-wise accuracy - An analysis of performance for each land cover category.
- Confusion matrix - To visualize misclassifications and identify patterns in errors.

Results

The experiment yielded significant results that underscored the importance of specific configurations and preprocessing techniques in image classification tasks. Initially, the model achieved an accuracy of 76.8% in training and 13% in validation, showcasing the challenges inherent in handling multispectral data. However, through iterative optimizations, including normalization of the spectral bands as per EuroSAT FAQ guidelines, accuracy improved substantially. Specifically:

- In the 50-50 split training accuracy reached 98.25%, validation accuracy reached 96.30%, while testing accuracy reached 98.40%.
- In the 80-20 split training accuracy reached 98.42%, validation accuracy reached 97.73%, while testing accuracy reached 98.83%.

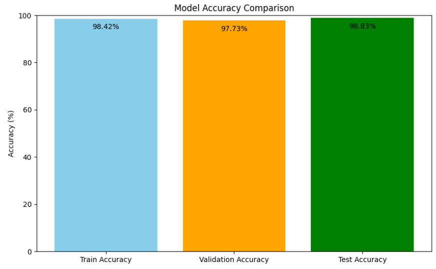


Figure 2: Accuracy comparison between training, validation, and testing for the split 80-20.

These results indicate that the applied methodologies successfully enhanced the model's capability to generalize across unseen data.

Final Considerations

The HybridModel architecture, with its fusion of convolutional and transformer layers, proved to be well-suited for the land cover classification task. The incorporation of

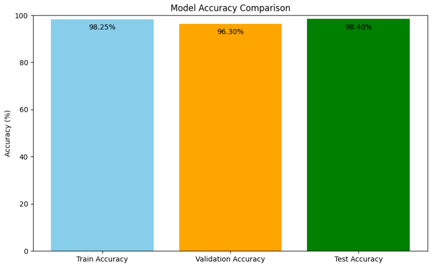


Figure 3: Accuracy comparison between training, validation, and testing for the split 50-50.

multi-band data provided richer information, enhancing the model's ability to distinguish between land cover types.

Key takeaways from the project include:

- The importance of leveraging the full spectral range available in the dataset.
- The efficacy of hybrid architectures in combining local and global feature extraction.
- The impact of rigorous preprocessing and augmentation on model performance.

Building on the current study, several avenues for future research and development could be explored to further enhance the performance and applicability of the model. Investigating more sophisticated design choices for the HybridModel architecture could significantly improve performance. Furthermore, evaluating the model on larger, more diverse datasets could demonstrate its scalability and generalization capabilities. Another future work could be the integration of domain-specific knowledge into the model pipeline in order to refine its predictive power.