Satellite Image Classifier

Raynold Yu, Anastan Gnanapragasam, Matthew Stebelsky, Adriel De Vera

Agenda

1. Goal of Project and Approach
2. Dataset
3. Frameworks
4. Machine Learning Algorithms Implemented
5. Results and Comparative Analysis
6. Methods to Improve the Model
7. Bibliography

Goal of Project

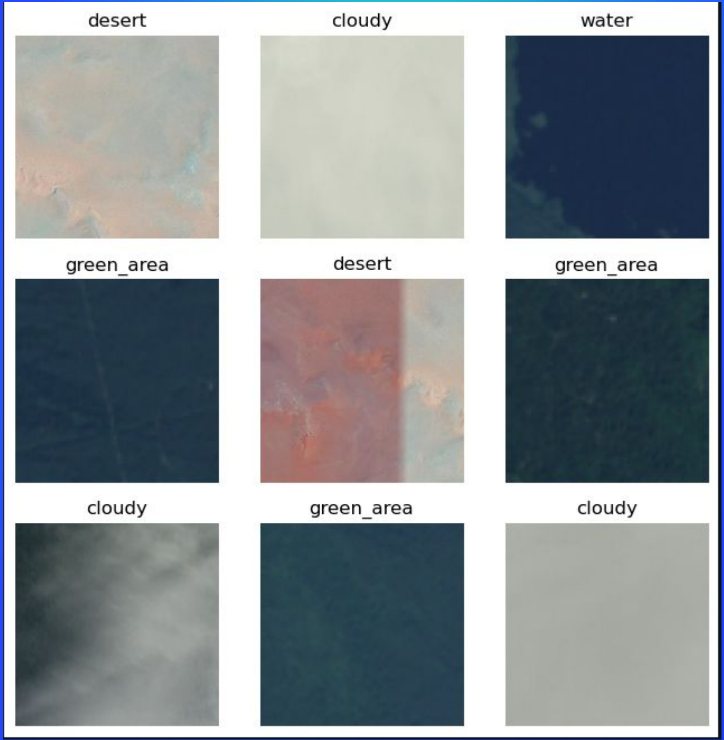
Leverage our technical understanding of Machine Learning Algorithms to accurately classify satellite images such as cloudy areas, deserts, grasslands, and bodies of water.

Specific Problem Addressed:

- There is a critical need to use the potential capabilities of satellite imagery in today's data-rich landscape
- Our focus seeks to contribute to comprehensive land cover analysis, aiding the analysis of ecosystems and climate change assessment for issues like desertification trends
- Promote data-driven decision-making for sustainable development and informed planning in various sectors

Use cases include governments, environmental agencies, humanitarian organizations, and industries reliant on accurate land cover information.

Approach



Standardize Dataset

- Resize images to same pixel size and resolution
- Normalize colour scales
- Remove outliers/corrupted images

Leverage Supervised Learning Methods

- Deep learning models, are well-suited for capturing complex image data
- Transfer learning can improve generalization and efficiency
- Random Forests prevents overfitting by averaging multiple decision trees and enables analysis of non-linear, intra-image relationships

Dataset:

The "Satellite Image Classification" kaggle dataset is a collection of satellite images from sensors and Google Maps. Organized in 4 classes;

Cloudy	Desert	Green Area	Water

The whole dataset has 5631 images with jpg format, each class has approx 1300 Images.

The dataset aims to support the development and testing of algorithms for automatic interpretation of remote sensing (RS) images by providing large-scale and practical image datasets for further data-driven research.

Manual Preprocessing Steps Taken:

- Resize image data to a uniform resolution size of 64 x 64 (Both models)
- Data Manipulate image data (rotate, flip, zoom) to increase dataset size and variability (CNN model)
- Convert grayscale images to RGB to standardize image channels for model (Random Forest Model RFM)
- Flatten the image matrices to one-dimensional arrays to prepare for model ingestion. (RFM)
- Detect and remove the alpha channel from images to maintain consistency in data format (RFM)

Machine Learning Algorithms Implemented

Random Forest Classifier

- Combines output of multiple decision trees
- Features:
  - Pixel values from resized images
  - Colors are normalized to improve model reliability and convergence
- Labels:
  - Categorical Biome Classifications
  - Converted labels to integers using label encoding
- Data Split:
  - 80% training set and 20% test set
- Model Evaluation:
  - Accuracy Score to quantify performance
  - Confusion Matrix to see accuracy score per label
  - Classification Report that reports the F1 score, precision and recall
- Visualization:
  - ROC Curves to show trade-off between true positive and false positive rate

Why is the Dataset is Appropriate?

- Each class size is balanced resulting in a fair training process
- Random forest performs well since each class has a large range of data sets
- Dataset is large enough to prevent overfitting
- Dataset includes images of biomes which are relevant to the scope of this project

Benefits and Drawbacks of Random Forest:

- Benefits:
  - More flexible, less prone to overfitting
  - Can model complex relationships between features that are non-linear
- Drawbacks:
  - Model as a whole is black box, making it hard to understand model decisions
  - Can be biased toward the majority class (Reconciled in data selection)

Machine Learning Algorithms Implemented

Convolutional Neural Network (CNN)

- Features:
  - Analyzes small sections of the input image (convolutional layers)
  - Consolidates the information gathered (max-pooling layers)
  - Makes decisions about the image content (fully connected layers)
- Data Split:
  - 80% training set, 10% validation set, and 10% test set
- Model Evaluation:
  - Accuracy Score to quantify performance
  - Confusion Matrix to visualize model's performance per class
  - Classification Report that reports metrics such as precision, recall, and F1-score
- Visualization:
  - Loss and Accuracy curves to track model's training progress
  - Histogram of predictions to visualize class distribution in test data

Benefits and Drawbacks of CNN:

- Benefits:
  - Robust to variations in position, scale, and orientation of objects within images.
  - Use parameter sharing - reduces number of parameters and memory requirements
  - Capture hierarchical representations of features
  - Can utilize parallel processing to accelerate training and inference
  - Utilize weight sharing across different spatial locations of the input image

- Drawbacks:
  - Requires large amounts of labeled data for effective training
  - Computationally intensive, requiring powerful hardware resources
  - Prone to overfitting with limited data or complex model architecture

Results and Comparative Analysis

Convolutional Neural Network

- Accuracy score of approximately 85.46%, indicating a reasonable performance by the model
  - The test loss was recorded at 0.3165, which suggests the model is generally fitting well, but still room for improvement
- Classification Report
  - Precision
    - Low across all classes, indicating that the model is not very accurate in classifying images as true positives
  - Recall
    - Generally low scores again, with ‘desert’ having the highest with 0.31, and ‘water’ having the lowest at 0.22
  - F1 Score
    - Weak across all classes, indicating an imbalance in precision and recall
  - Macro Avg
    - All equal to 0.26, showing that the model is uniformly underperforming across all classes without a significant bias toward any class
- Confusion Matrix
  - Indicates that the 'water' class is the most challenging for the model to classify correctly
- Histogram of Predictions
  - Shows a bias in the number of predictions for 'green_area' and ‘desert’ class, which may explain the higher recall for those

Suggestions for Improvement

- Enhancing the CNN's ability to distinguish 'water' images could significantly improve overall performance
- Data augmentation, hyperparameter tuning, or more advanced models could be employed to balance precision and recall
- Review the training process to check for data imbalance, insufficient feature extraction, or over/under fitting

Results and Comparative Analysis Continued:

Random Forest Classifier

- Accuracy score of 95.03% indicating that the model is accurately classifying images in the test dataset
  - Not likely to be overfitted due to balanced class size and nature of random forest classifiers
- Classification Report
  - Precision
    - High Precision scores across all classes with a perfect score of 1.0 for ‘Desert’ class
  - Recall
    - ‘Cloudy’ class had a 0.99 score making it the highest Recall score across all classes
    - ‘Water’ class had a 0.88 score making it the lowest which suggests room for improvement
  - F1 Score
    - Solid scores showing balanced precision-recall tradeoff
  - Macro Avg
    - Is 0.95 which shows the model is performing consistently well across all classes without significant bias towards any classes
- Confusion Matrix
  - Classifier struggled the most to classify the ‘Water’ class while it accurately classified other classes
- ROC Curve
  - ROC curve highlights high AUC values among all classes indicating the model is effective in distinguishing biomes
  - ROC AUC score is 0.99 which shows a near perfect score which shows that the binary classifier is accurately able to separate between true positive and false positive

Methods to Improve the Model:

- Hyperparameter tuning:
  - Testing different number of trees
    - larger number of trees increase model accuracy but may reach plateau
  - Depth of each tree
    - The depth controls the models complexity
    - Would require testing to identify optimal depth to prevent over/under-fitting
- Cross-validation:
  - Use k-fold cross-validation to test model’s performance across different subsets
  - Can be used to identify optimal tree count for the model
- Data Augmentation:
  - Apply transformations such as flip and rotate to help the model generalize unseen data more efficiently
- Transfer Learning:
  - Utilize transfer learning by starting with a model pre-trained on a large dataset and fine-tune it to a specific task
- Parameter Testing
  - Use tools like GridSearch to systematically explore different hyperparameter combinations, optimizing model performance by selecting the best set of hyperparameters

Bibliography

- N. Jmour, S. Zayen and A. Abdelkrim, "Convolutional neural networks for image classification," 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), Hammamet, Tunisia, 2018, pp. 397-402, doi: 10.1109/ASET.2018.8379889.
- Pritt, Mark & Chern, Gary. (2017). Satellite Image Classification with Deep Learning. 1-7. 10.1109/AIPR.2017.8457969.
- Reda, Mahmoud. “Satellite Image Classification | Kaggle.” *Kaggle: Your Machine Learning and Data Science Community*, 21 Aug. 2021, https://www.kaggle.com/datasets/mahmoudreda55/satellite-image-classification/data.
- Rivera, Antonio & Rolando, Julio & Cabezas, Coello & Ronald, Emer & Solórzano, Rosales & Andrés, Carlos & Herrera, Mancheno & Guillermo, Gregory & Andrade, Cuesta & Elena, Carmen & Cabrera, Mantilla. (2023). Satellite Image Using Image Processing and Machine Learning Techniques: applications to agriculture, environment and mining.. European Chemical Bulletin. 1718-1734. 10.31838/ecb/2023.12.1.256.