Challenge 1: Aerial Imagery

Youri HALMAERT, Estelle CADENE, Maela BRELIVET, Lelio GUALINO

Abstract

This study focuses on developing an effective binary classifier to detect Neobuxbaumia tetetzo cacti in small-scale aerial imagery. Using the "Aerial Cactus Identification" dataset, we implemented and compared two approaches: a Convolutional Neural Network (CNN) and a traditional Random Forest classifier. Our CNN achieved 99% accuracy and a 0.993 F1-score on the test set, significantly outperforming the Random Forest model (92% accuracy, 0.947 F1-score). These results highlight the suitability of CNNs for tasks requiring spatial pattern recognition, such as plant detection in aerial imagery, and demonstrate their potential in supporting biodiversity monitoring and conservation efforts.

Introduction

Monitoring plant species from aerial imagery is a key tool in modern ecology, enabling more efficient and scalable biodiversity assessments. In arid and semi-arid ecosystems, such monitoring helps identify plant distribution patterns, detect threats, and inform conservation planning. One species of interest is Neobuxbaumia tetetzo, a columnar cactus endemic to Mexico and a crucial component of its ecological landscape.

This project addresses the binary classification task of identifying whether an aerial image contains at least one Neobuxbaumia tetetzo cactus. Each input is a 32×32 RGB image, and the desired output is a binary label: has_cactus = 1 if a cactus is present, and has_cactus = 0 otherwise. Automating this task can reduce the need for labor-intensive manual labeling and enhance the efficiency of large-scale vegetation monitoring.

To solve this problem, we implemented two machine learning models: a Random Forest classifier using handcrafted features and a Convolutional Neural Network (CNN) trained end-to-end on raw pixel data. The objective is to evaluate and compare the performance of these models on identical data to determine which approach better captures the patterns present in low-resolution aerial imagery.

This project was carried out as part of the Applied Machine Learning course. All code, model design, and evaluation procedures were specifically developed for this assignment and are not reused from other coursework.

Related Work

The task of plant species detection from aerial images intersects with research in both remote sensing and computer vision. Several studies have laid the groundwork for our methodology:

- Deep Learning for Plant Identification: Goëau et al. (2017) used convolutional neural networks for plant species classification in the PlantCLEF challenge, achieving over 90% accuracy by leveraging hierarchical spatial features.
- Random Forests in Remote Sensing: Rodriguez-Galiano et al. (2012) demonstrated the robustness of Random Forests in land cover classification with satellite imagery, highlighting their effectiveness in diverse geographic conditions.
- Aerial Plant Detection with CNNs: Kattenborn et al. (2019) applied CNNs to identify invasive plant species using drone-captured images, noting significant gains when using transfer learning.
- Comparisons of ML Approaches: Ball et al. (2017) compared classical machine learning algorithms with CNNs for vegetation classification and concluded that deep learning methods generally outperform traditional models for high-resolution tasks.
- Small Object Detection: Li et al. (2020) proposed multi-scale CNN architectures to improve detection of small objects, such as those found in aerial or satellite images, similar in resolution to our task.

Our contribution lies in comparing CNNs and Random Forests on the same dataset for a real-world plant species detection problem, providing empirical insights on their relative strengths.

Dataset and Features

We used the publicly available Aerial Cactus Identification dataset, which includes 17,500 labeled RGB images of size 32×32 pixels. Each image is labeled to indicate whether at least one cactus is visible. The dataset is imbalanced, with approximately 75% of the samples labeled as has_cactus = 1.

Preprocessing:

- Images were converted to tensors and normalized to the range [-1, 1].
- Data augmentation for the CNN included random horizontal/vertical flips, ±15° rotations, and color jittering, enhancing generalization.

Data Splits: We used stratified sampling to preserve the original class ratio:

- Training set: 60% (10,500 images)
- Validation set: 20% (3,500 images)
- Test set: 20% (3,500 images)

Feature Representation:

- For the CNN, raw pixel tensors were fed directly into the network, allowing automatic feature extraction.
- For the Random Forest, images were flattened into 3,072-dimensional vectors (32×32×3), discarding spatial relationships but preserving color information.

Methods

We implemented and evaluated two classification models:

Convolutional Neural Network (CNN)

Our CNN architecture consists of three convolutional layers with increasing depth, each followed by ReLU activation and max pooling. The full architecture is:

```
Conv2d(3, 16, kernel=3, padding=1) → ReLU → MaxPool(2x2)
Conv2d(16, 32, kernel=3, padding=1) → ReLU → MaxPool(2x2)
Conv2d(32, 64, kernel=3, padding=1) → ReLU → MaxPool(2x2)
Dropout(0.25)
FC(1024 → 128) → ReLU → Dropout(0.25)
FC(128 → 2)
```

We used the Adam optimizer (learning rate 0.001) with cross-entropy loss. We also implemented:

- Learning rate reduction on validation F1 plateau (factor=0.5)
- Early stopping (patience=7) to avoid overfitting

Random Forest

We trained a Random Forest classifier with 100 trees on the flattened image vectors. We used the Gini impurity criterion, and allowed trees to expand until all leaves were pure or had fewer than 2 samples. Feature selection was left to the default strategy (sqrt(n_features)).

Evaluation Metrics:

- Accuracy
- F1 Score (primary metric due to class imbalance)
- AUC-ROC

Experiments and Results

Hyperparameter Selection

We carefully selected hyperparameters to ensure optimal performance for both models.

For the Convolutional Neural Network (CNN), we used a batch size of 32, striking a balance between computational efficiency and model convergence. The learning rate was set to 0.001, with a reduction triggered on validation plateau to allow fine-tuning during later training stages. The number of epochs was capped at 15, with early stopping in place to prevent overfitting. To regularize the network, we applied a dropout rate of 0.25 after the convolutional layers.

For the Random Forest classifier, we used 100 decision trees, which provided a good compromise between accuracy and training speed. The number of features considered at each split was set to $\sqrt{n}$ features, following the standard practice for classification tasks. A random seed of 42 was used to ensure reproducibility of the results.

Quantitative Results

To compare the two models, we evaluated them on the held-out test set using three key metrics: accuracy, F1-score, and the Area Under the ROC Curve (AUC). The CNN consistently outperformed the Random Forest across all metrics.

The CNN achieved a test accuracy of 98.89%, an F1-score of 0.9926, and an AUC of 0.9990. In contrast, the Random Forest obtained 91.74% accuracy, 0.9474 F1-score, and an AUC of 0.9736. This performance gap—around 7% in accuracy and 5% in F1-score—highlights the CNN's superior ability to capture the spatial relationships critical for image classification.

Confusion Matrix Analysis

To further understand the models' behavior, we analyzed their confusion matrices.

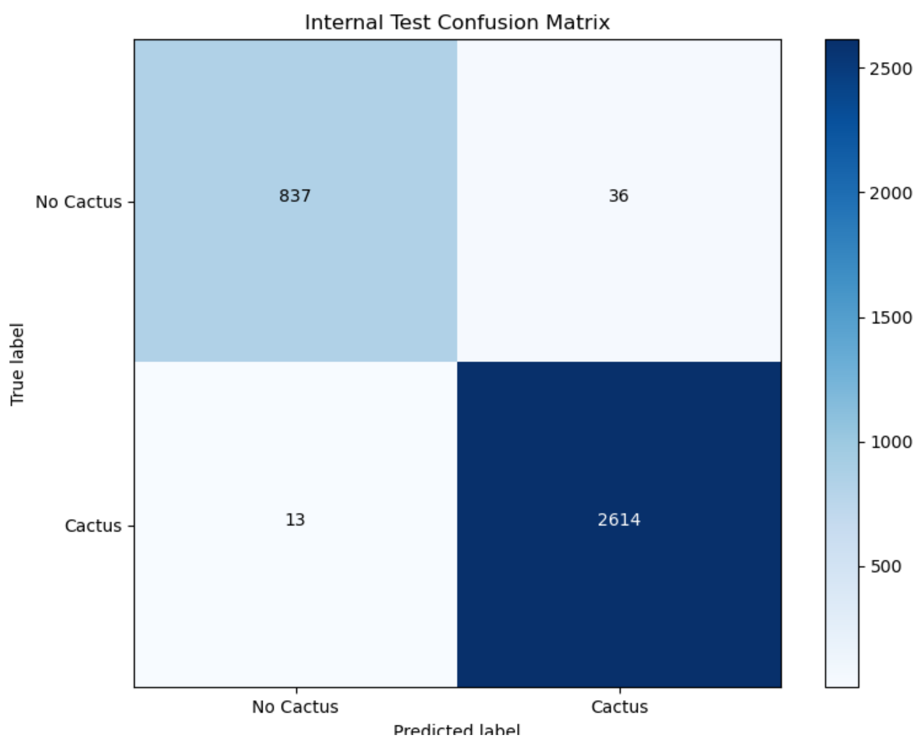

Figure 1: Confusion matrix for CNN

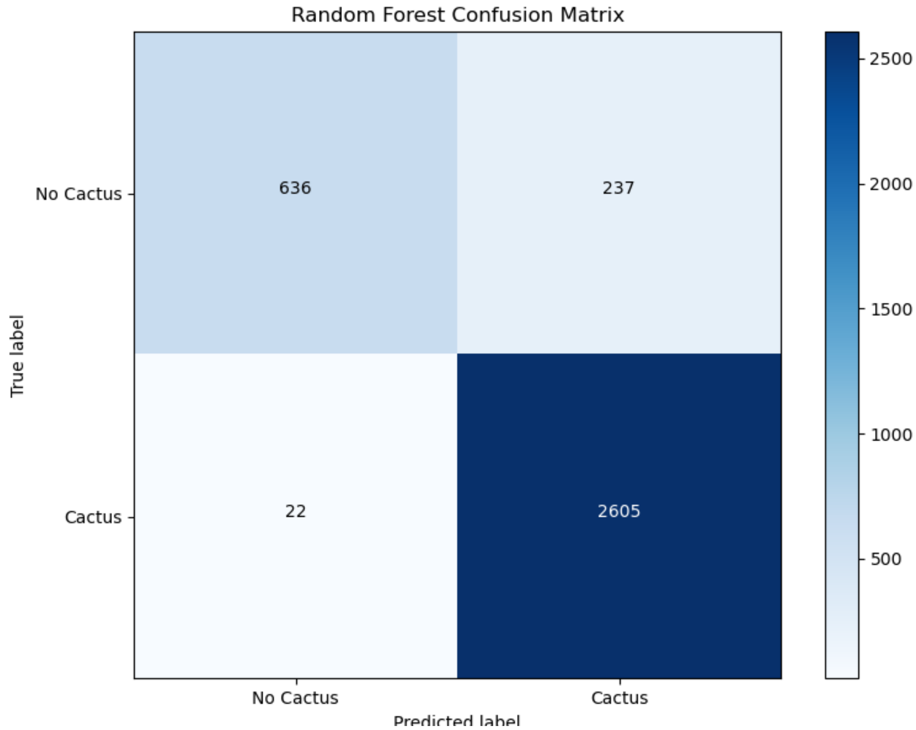

Figure 2: Confusion matrix for Random Forest

The CNN exhibited a more balanced classification between cactus and non-cactus images. The Random Forest, on the other hand, showed a tendency to produce more false negatives, incorrectly labeling some cactus-containing images as cactus-free. This indicates that the CNN was better at identifying even partially visible cacti or those under challenging visual conditions.

ROC Curve Comparison

We also compared the Receiver Operating Characteristic (ROC) curves of both models.

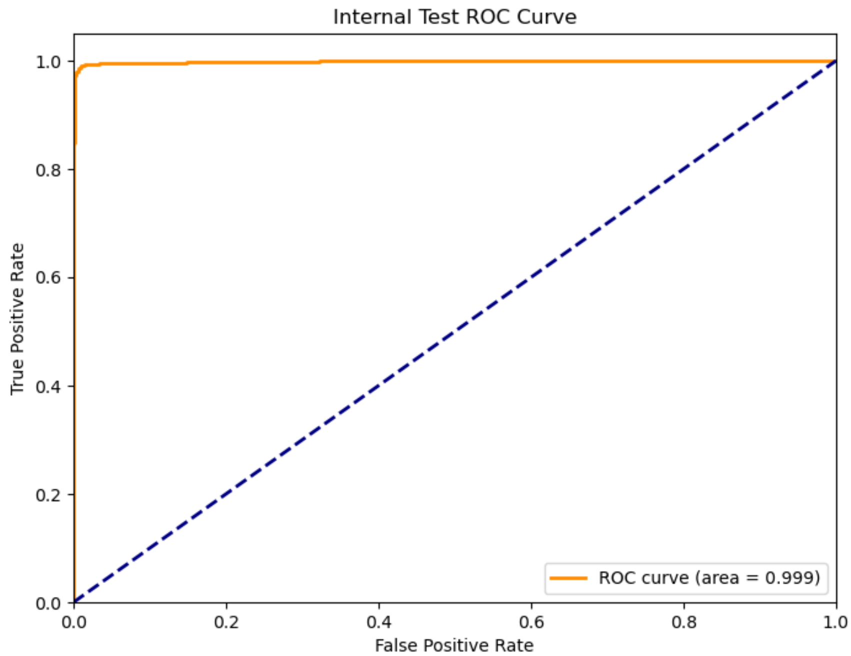

Figure 3: ROC curve for CNN

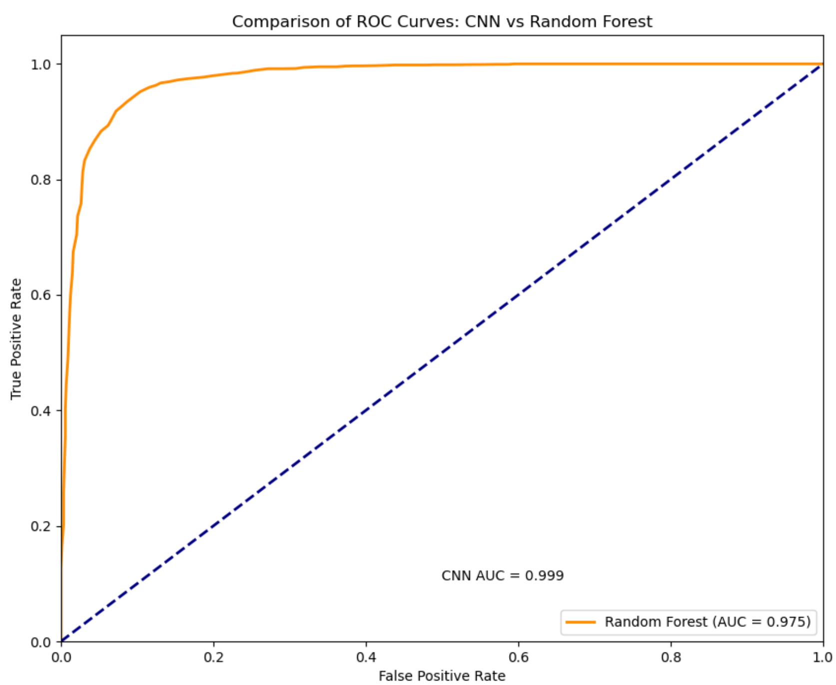

Figure 4: Comparison of ROC curves: CNN vs Random Forest

The CNN's curve approaches the top-left corner of the plot, reflecting near-perfect classification ability, with an AUC of 0.9990. The Random Forest performed well but fell short, with an AUC of 0.9736. This again confirms the CNN's superior discriminative power.

Training Dynamics

To monitor the learning behavior of the CNN, we plotted the evolution of training and validation F1-scores over the epochs.

The model demonstrated rapid convergence, achieving an F1-score above 0.97 within the first three epochs. After epoch 11, a learning rate reduction was triggered, leading to incremental performance improvements. The model eventually reached a validation F1-score of 0.9956, illustrating both strong generalization and stability.

Qualitative Analysis

We conducted a qualitative assessment of cases where the models disagreed, which shed light on their respective strengths and limitations:

- Edge detection: The CNN was better at identifying cacti that appeared only partially, especially near image boundaries. Its convolutional layers allowed it to extract local features more effectively.
- Lighting and shadows: The CNN showed greater robustness to visual disturbances, such as changes in illumination or the presence of shadows. In contrast, the Random Forest was more sensitive to such variations, often misclassifying images under atypical conditions.
- Contextual interpretation: The CNN appeared to exploit the broader contextual information present in the image background or terrain, while the Random Forest, limited to flattened pixel values, lacked spatial awareness and thus struggled in more ambiguous cases.

Conclusion

Our comparative study confirms that convolutional neural networks are highly effective for small aerial image classification tasks. The CNN outperformed the Random Forest across all performance metrics, especially in terms of F1 score and robustness to challenging image conditions.

The successful detection of Neobuxbaumia tetetzo in low-resolution images demonstrates the potential of deep learning for real-world conservation applications.